LifeLines Documentation

LifeLines Version 3.1.1

Thomas T. Wetmore , IV

COLLABORATORS

	TITLE : LifeLines Documentation	

ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY	Thomas T. Wetmore , IV	June 29, 2026	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1 Users' Manual **1**

1.1 INTRODUCTION . . . 1
1.2 INSTALLATION . . . 2
1.3 STARTING LIFELINES AND CREATING DATABASES . . . 2
1.4 INTRODUCTION TO GEDCOM . . . 3
1.5 MAIN MENU . . . 6
1.6 CODESET . . . 7
1.7 ENTERING THE FIRST PERSON . . . 7
1.8 SCREEN EDITORS AND ENVIRONMENT VARIABLES . . . 8
1.9 BASICS OF BROWSING . . . 9
1.10 IDENTIFYING A PERSON OR LIST OF PERSONS TO BROWSE . . . 9
1.11 ZIP IDENTIFYING A NEW PERSON . . . 10
1.12 BROWSE DISPLAY BASICS . . . 11
1.13 PERSON BROWSE SCREEN . . . 11
1.14 LIST BROWSE SCREEN . . . 13
1.15 FAMILY BROWSE SCREEN . . . 15
1.16 TANDEM PERSON BROWSE MODE . . . 16
1.17 TANDEM FAMILY BROWSE MODE . . . 18
1.18 PEDIGREE BROWSE MODE . . . 19
1.19 SEARCH MENU . . . 20
1.19.1 Search Patterns . . . 21
1.20 ADD OPERATION . . . 21
1.21 DELETE OPERATION . . . 22
1.22 CODESET CONVERSION . . . 22
1.23 MISCELLANEOUS UTILITIES . . . 24
1.24 IMPORT ERRORS . . . 26
1.25 HANDLING SOURCE, EVENT AND USER-DEFINED RECORDS . . . 28
1.26 FAMILY STRUCTURE AND MERGING PERSONS AND FAMILIES . . . 32
1.27 LINKING RECORDS TOGETHER AND USING THE REFN FEATURE . . . 33
1.28 LIFELINES REPORTS . . . 34

1.29 System and User Properties 34
1.30 Other Executables 38
1.30.1 dbverify 38
1.30.2 btedit 39
1.30.3 llexec 39

Chapter 1

Users' Manual

1.1 INTRODUCTION

LifeLines is a genealogy program that runs on UNIX systems. It maintains genealogical records (persons, families, sources, events and others) in a database, and generates reports from those records. There are no practical limits on the number of records that can be stored in a LifeLines database, nor on the amounts or kinds of data that can be kept in the records. LifeLines does not contain built-in reports. Instead it provides a programming subsystem that you use to program your own reports and charts (which has a separate manual -- the lifelines report manual). The programming subsystem also lets you query your databases and process your data in any way. LifeLines uses the terminal independent features of UNIX to provide a screen and menu based user interface.

The database is created by entering data with LifeLines. Data can also be imported from GEDCOM files or exported to GEDCOM files.

LifeLines is a non-commercial, experimental system that is use at your own risk software. I developed LifeLines for personal use and shared it with friends. Enough of a demand arose through word of mouth and internet, that I have made the LifeLines source code and other information freely available under an MIT-style license reproduced below:

" Copyright (c) 1991-1999 Thomas T. Wetmore IV Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions: The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software. THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE. "

The source code, documentation and a collection of report scripts are located at https://lifelines.github.io/lifelines/. You can also find binary kits for some platforms. If you are a developer and wish to contribute enhancements, please obtain a github account and contact Marc Nozell who is currently managing the project.

Note
Prior to 1999, Lifelines was available on the ftp sites, ftp.cac.psu.edu and hoth.stsci.edu. Please use github instead.
From 1999 to 2015, Lifelines was available from sourceforge. Please use github instead.

Other sources of information include:

lifelines homepage	https://lifelines.github.io/lifelines/

lifelines FAQ	https://lifelines.github.io/lifelines/faq.html.
LINES-L mailing list	http://listserv.nodak.edu/cgi-bin/wa.exe?A0=lines-l.

1.2 INSTALLATION

You may be installing LifeLines from a source distribution package or as an executable program already prepared for your UNIX or MS-Windows system. The source distribution comes with the readme, build script and make files necessary to build LifeLines. Follow the instructions in the readme file. A number of executables are built which can be put it in a directory in your execution path. If you get the program in executable form, follow whatever instructions came with it. The following executables are included:

btedit

a low-level editor for advanced debugging of broken LifeLines databases.

dbverify

a utility program to verify LifeLines databases.

llexec

a stripped down version of the LifeLines program, without the user interface for non-interactive processing of report programs

llines

the LifeLines program with full user interface

The source distribution package also contains documentation and some LifeLines programs to demonstrate the capabilities of the report language. Included with these in the reports directory is a brief overview of the reports in the file index.html.

1.3 STARTING LIFELINES AND CREATING DATABASES

You normally start LifeLines with the command: **`llines`** ***`database`*** where database is the name of a LifeLines database. If LifeLines finds the database, LifeLines opens the database and takes you to the program's main menu. If the database doesn't exist, LifeLines asks whether it should create it, and if you answer yes, does so. You may create any number of databases, but only one can be accessed by LifeLines at a time.

The full command line interface to LifeLines is:

```
llines [-acdfiklnortuwxzCFI][database]
```

The following options are supported:

-C	Specify configuration file location (e.g. -C/home/bill/lifelines/.linesrc2) see the section on System and User Properties below
-F	Finnish option (only available if so compiled)
-I	Specify a user property (e.g. -ILLEDITOR=gvim)
-a	log dynamic memory operation (for debugging)
-c	supply cache values (eg, -ci400,4000f400,4000 sets direct indi & fam caches to 400, and indirect indi & fam caches to 4000)
-d	debug mode (signal protection disabled for convenience with breakpoints)
-f	force open the database - use only in emergency
-i	open database with immutable access (no protection against other access -- for use on read-only media)
-k	always show keys (normally keys are suppressed if REFN available)

-l	lock (-ly) or unlock (-ln) a database for use with read only media (access to a locked database is treated as immutable)
-n	do not use traditional family rules
-o	Specify program output filename (eg, -o/tmp/mytests)
-r	open database with read-only access (protect against other writer access)
-t	trace function calls in report programs (for debugging)
-u	specify window size (eg, -u120,34 specifies 120 columns by 34 rows)
-w	open database with writable access (protect against other writer or reader access)
-x	execute a single lifelines report program directly
-z	Use normal ASCII characters for drawing lines in user interface rather than the vt100 special characters.

the `-o` option specifies the initial filename to use for output when running reports. It only applies to reports run with the `-x` option. This option has no effect on interactively run programs.

The `-r` option opens the database with read-only access. When in this mode LifeLines will not let you modify the database; no other operations are affected. The `-w` option opens the database with writable access. If the database cannot be opened with the requested mode LifeLines quits immediately. When you open a database with neither the `-r` or `-w` options, LifeLines first tries to open the database with writable access; if not possible LifeLines then tries to open the database with read-only access; and if this is not possible LifeLines quits. A LifeLines database may be open simultaneously by any number of programs with read-only access; however, if a database is open by a program with writable access, then it cannot be opened by any other other program.

In rare situations the read/write mode mechanism can fail; when this happens a database may appear unopenable. If this happens use the `-f` option to force open the database; this will open the database and reset the mode mechanism. This is a dangerous feature; you can use it to open the same database with writable access more than once; the results are unpredictable and generally disastrous.

The multiuser protection supplied by this reader/writer access mechanism is provided via a flag setting in the database, so both read-only and writable access actually alter the database (read-only access only alters the value of this flag). For truly read-only access, e.g., for use with read-only media, the best solution is to lock (-ly) the database before copying it to the read-only media. This annotates the database itself as being for immutable access. Alternatively, to use a database already on read-only media and not so annotated, use the immutable (-i) flag.

By default lifelines supports a traditional family concept, that is, each family has at most one father and one mother. The -n flag relaxes this restriction. However, not all the code in lifelines supports these relaxations. For example, the default family browse screen will only display two parents, however by switching to one of the gedcom modes of displaying the family you can see all the data.

If you don't give the name of a database on the command line, LifeLines will prompt you for it. If the name you supply is an absolute pathname or a relative pathname it is used as the path to the database. If you provided a simple filename and you use the `LLDATABASES` variable or user options (described later), LifeLines will search for the database in the directories named in the variable; this can be very convenient. If `LLDATABASES` is not set the filename you enter is looked for in the current working directory.

If you would like to choose a database from a list of existing ones, enter a single question mark and press return when LifeLines prompts you for the database name. LifeLines will then display a list of all databases that it can find, and you may select one from the list.

1.4 INTRODUCTION TO GEDCOM

LifeLines records are stored in GEDCOM format; you organize, edit and maintain your data in this format. GEDCOM is a standard that defines a file format for moving genealogical data between computer systems. LifeLines has adopted this format for structuring the records in its databases. This approach provides a structured yet flexible method for storing all the data you wish to record. There are few restrictions on the format, amount or type of information you may store in a LifeLines database.

GEDCOM is defined at two levels. At the syntactic level GEDCOM is a simple set of rules for organizing and structuring data into records, with no concern about the types of records, types or formats of information in the records, or the relationships between records. At the semantic level GEDCOM adds an additional set of rules that specify what record types are allowed,

how records must be structured, how data within the records must be identified and formatted, and what specific relationships between the record types are allowed. In principle there can be multiple semantic versions of GEDCOM, though in practice there is only one, lineage-linked GEDCOM. Unfortunately this semantic version of GEDCOM is poorly defined, and every genealogical system has interpreted it in different ways.

LifeLines uses GEDCOM primarily at the syntactic level, though it does impose a few of the generally accepted lineage-linked semantic restrictions. This means some important things. It means that you can store all your genealogical data in your LifeLines database, and that you have wide freedom in how you choose your own conventions for structuring and formatting your data. But it also means that the way you store data in your databases can be different from the way that someone else stores their data. This can be a problem if you share data with others or share report programs with other LifeLines users. It is recommended to use GEDCOM lineage-linking conventions wherever possible. LifeLines allows you to export it's database to a GEDCOM file for archival purposes or to exchange data with others. You can also import data from GEDCOM files into a LifeLines database. (See Miscellaneous Utilities for details.)

LifeLines does not use forms or screens to guide you through entering or changing data. Instead you use a screen editor and directly edit the data records. This requires you to understand the GEDCOM format, and be able to edit data in GEDCOM format, before you can use LifeLines. The GEDCOM format is quite simple; this introduction will provide all you need to know about GEDCOM in order to use LifeLines.

Here is an example GEDCOM person record:

Example 1.1 GEDCOM person record

```
0 @I25@ INDI
1 NAME Thomas Trask /Wetmore/ Sr
1 SEX M
1 BIRT
  2 DATE 13 March 1866
  2 PLAC St. Mary's Bay, Digby, Nova Scotia
  2 SOUR Social Security application
1 NATU
  2 NAME Thomas T. Wetmore
  2 DATE 26 October 1888
  2 PLAC Norwich, New London, Connecticut
  2 AGE 22y
  2 COUR New London County Court of Common Pleas
  2 SOUR court record from National Archives
1 OCCU Antiques Dealer
1 DEAT
  2 NAME Thomas Trask Wetmore
  2 DATE 17 February 1947
  2 PLAC New London, New London, Connecticut
  2 AGE 80y11m4d
  2 CAUS Heart Attack
  2 SOUR New London Death Records
1 FAMC @F11@
1 FAMS @F6@
1 FAMS @F12@
```

A GEDCOM record is made up of lines. Each line has a level number and a tag, and most lines have a value following the tag. The first line in every record has a cross-reference index between the level number and the tag.

Level numbers allow data to be structured to any degree of detail; lines with higher level numbers expand on lines with lower numbers. Each record begins at level 0, and each deeper level increments the level by one. LifeLines does not restrict the structuring depth. Tags are uppercase (by convention) code words that specify the kind of information on the line or on the higher numbered lines that follow. The information after the tag, if any, is the value of the line.

The first line in a record indicates its type. There are four fixed record types in LifeLines databases: person, family, source and event. The first, 0 level line in these records have tags `INDI`, `FAM`, `SOUR` and `EVEN`, respectively. Besides these record types, you may create your own record types by using any other tag on the 0 level line of a record. The lines that begin records are the only level 0 lines used in LifeLines. Each level 0 line has a cross-reference index between the level number and the tag. This

index is the record's internal reference key; other records may refer to this record by using this index. Cross-reference indexes are bracketed by @ characters.

The first line in the example record has the `INDI` tag, identifying it as a person. The cross-reference index value, I25, can be used by other records to refer to this record.

The second line in the example has the person's name. Each person record in a LifeLines database must have at least one `1 NAME` line, and its value must be in GEDCOM name format. This format allows names to be as long as needed, but the surname, which may be placed anywhere in the name, must be separated from the rest of the name by one or two slashes. For example:

Example 1.2 Example of NAME formats

```
1 NAME John/Smith
1 NAME John /Smith/
1 NAME John/Smith/Jr.
```

The second slash is required only if name elements follow the surname. White space is optional before the first slash and after the second. If you don't know a person's surname, or the person doesn't have a surname, you may use / or // or no slashes at all. For example:

Example 1.3 Example of searching on NAMEs

```
1 NAME Mary//
1 NAME Mary/
1 NAME Mary
```

are all ways to enter a person named Mary with no known surname. A person may have any number, including zero, given names and/or initials. A LifeLines person record may have any number of `1 NAME` lines, though the person will be displayed with the first name value only. Persons are indexed under all their names, however, so you will be able to search for persons by any of their names.

The next line in the example gives the person's sex. LifeLines doesn't require a `1 SEX` line, but you should include it. The value of the line should be **`M`** or **`F`** if the sex is known; it can be left blank or set to **`U`** or **`?`**, say, if not known. A person must have a `1 SEX` line with a value of either **`M`** or **`F`** before he or she can be made a spouse or parent in a family.

The example record also contains three events: birth, naturalization, and death. An event begins with a level 1 line whose tag indicates the event type. For example, `BIRT` is the tag for a birth event.

Events usually have at least a `2 DATE` and a `2 PLAC` line and often a `2 SOUR` line. The `DATE` and `PLAC` lines give the date and place of the event. The value of a LifeLines `DATE` line is free format, though LifeLines will try to parse it for specific day, month and year information. The value of a `PLAC` line is usually a comma-separated list of geopolitical units, starting with the most specific, ending with the most general. The `SOUR` line indicates the source of information about the event. The `SOUR` line can be the root of a full description of the source, or the value of the `SOUR` line can be a cross-reference key that refers to the source record that describes the source.

The naturalization event (with tag `NATU`) shows a few other lines. The `2 NAME` line gives the person's name as recorded in the source (only `1 NAME` lines must follow GEDCOM format). The `2 AGE` line gives the person's age at the time of the event. The `2 COUR` line indicates the court where naturalization occurred.

The final event is a death event (tag `DEAT`). The `2 CAUS` line gives the cause of death.

At the end of the record are three lines that refer to family records. A `1 FAMC` line refers to a family record that the person belongs to as a child; its value is the cross-reference index value of that family. A `1 FAMS` line refers to a family record that the person belongs to as a spouse or parent.

When using LifeLines to edit a person, you will not be able to edit the cross reference values on the `0 INDI`, `1 FAMC` or `1 FAMS` lines; these are maintained by LifeLines.

Here is an example family record:

Example 1.4 Example family record

```
0 @F6@ FAM
1 HUSB @I25@
1 WIFE @I26@
1 MARR
  2 DATE 31 March 1891
  2 PLAC New London, New London, Connecticut
  2 SOUR New London Vital Records
1 CHIL @I27@
1 CHIL @I17@
```

The `0 FAM` line assigns the family the cross-reference index of F6. The record contains `1 HUSB` and `1 WIFE` lines that refer to the two spouses/parents. The record also holds a marriage event (tag `MARR`) and two `1 CHIL` lines that refer to two children in the family. When editing family records, you cannot edit the `0 FAM`, `1 HUSB`, `1 WIFE`, or `1 CHIL` lines; these are maintained by LifeLines.

When you create new records for your database, you are free to invent tags and structure your data in any way you see fit. However, it is good practice to use standard GEDCOM tags and value formats. LifeLines does enforce a small set of conventions that you must obey. Within person records, LifeLines requires that you use `1 NAME` and `1 SEX` lines with their special meanings and value formats. Though not required, LifeLines assumes that you will use `1 BIRT`, `1 DEAT`, `1 CHR`, and `1 BURI` lines for birth, death, baptism and burial events, respectively. In family records, LifeLines assumes you will use the `1 MARR` event for marriage events. Within person records, you are not allowed to use `0 INDI`, `1 FAMC` or `1 FAMS` lines, since these are used to maintain linkage information. Within family records, you are not allowed to use `0 FAM`, `1 HUSB`, `1 WIFE` or `1 CHIL` lines.

The indentation shown in the examples is not part of GEDCOM format. When LifeLines prepares records for you to edit, however, it always indents the records, making them easier to read and understand. You do not need to follow this indentation scheme when you edit the records. Indentation is removed from the data before it is stored in the database.

1.5 MAIN MENU

After LifeLines opens an existing database, or creates a new one, it presents you with the main menu:

```
Please choose an operation:
   b Browse the persons in the database
   s Search database
   a Add information to the database
   d Delete information from the database
   p Pick a report from list and run
   r Generate report by entering report name
   t Modify character translation tables
   u Miscellaneous utilities
   x Handle source, event and other records
   Q Quit current database
   q Quit program
```

Select an operation by striking the proper selection letter.

The browse operation lets you browse the database and perform many operations on the data. The search operation provides some simple wildcard search capabilities, which lead into browsing particular records. The add operation lets you add new information, and the delete operation removes information. The report operations read report programs and generates output reports. The modify character translation tables operation changes the translation tables. The miscellaneous utilities operation provides such things as backup and restore. The handle source, event and other records operation gives you access to these three record types. The quit operation closes the database and returns to UNIX.

The browse operation deserves special mention, because it provides a rich environment for searching, viewing, adding, modifying, merging and deleting information in the database. You will find that you operate from the browsing modes most of the time. The operations are all described in later sections.

1.6 CODESET

After you have created a new database, and before you actually add any data to it, is the time to set the codeset to be used in the database.

The codeset (or character encoding, to use precise Unicode terminology) is the decision as to how letters will be represented by the computer. If you have only ever used English letters in computing, you may not have had to encounter this issue, because as it happens, the English letters (a-z and A-Z) are stored numerically in the same fashion in almost all codesets used by computers. However, in the field of genealogy, you are especially likely to meet letters outside of the English alphabet (for example, accented vowels).

You have fundamentally three choices as to what codeset to use in your database, listed below from easiest to most powerful.

First, you may leave it entirely unspecified. This will give the traditional lifelines behavior. This is really only suitable if either (a), you only use English (ASCII) data, or (b), you work in an environment which entirely uses the same 8-bit codeset (eg, a GNU/Linux box which is all ISO-8859-15), and you only run lifelines in English. If you use any non-English data on MS-Windows, this is not likely to be suitable, because the lifelines screens run in the console, but you are likely to use MS-Windows applications either for editing or for viewing output, and the MS-Windows console uses a different codeset from MS-Windows applications. Also, if you use lifelines in a different language than English, this may not be suitable, because the gettext message catalogs (for non-English interface) will not be converted into your codeset.

Second, you may specify a particular 8-bit codeset. Assuming that you have iconv and gettext installed (or you are using the MS-Windows version, which comes with these), you may specify any 8-bit codeset supported by iconv, and iconv supports quite many. A natural choice for Western European languages would be ISO-8859-1, or (for MS-Windows only) CP-1252. With this option, gettext language files will be converted to your codeset.

Third, you may specify the use of UTF-8. This is a Unicode encoding, and is by far the most powerful option. In fact, this is the only really convenient way to be able to store, for example, names in English, names in Russian, and names in Greek, all in the same database, in their native scripts (alphabets). In recent versions, lifelines has become more knowledgeable about handling UTF-8, so that, for example, upper & lower casing only work correctly with versions from 3.0.28 on.

To actually specify a codeset, enter it via the u(tility) o(ptions) page (which is documented below). From the main menu, in the English version, press u to reach the utility page, and then o to edit the user options. To set a codeset of, e.g., ISO-8859-1, enter this string on its own line, without the surrounding quotes: "codeset=ISO-8859-1". Or, to specify the use of UTF-8, "codeset=UTF-8".

Further information about codeset conversion is found in the later chapter of that name (for example, information about producing reports which make use of HTML entity names for non-ASCII characters).

1.7 ENTERING THE FIRST PERSON

Note: Before you add the first person to your database, you specify internal codeset (review the Codeset chapter for information).

Normally you add persons to the database from the browsing modes, but when entering the first person there is no one in the database to browse to. To add the first person to a LifeLines database, first select the add operation from the main menu. You will be prompted with the add menu (described later). Strike p to add a person. LifeLines creates a template of a GEDCOM person record, and puts you in a screen editor to edit the template. The default template is:

Example 1.5 Default person record template

```
0 INDI
1 NAME Fname /Surname/
1 SEX MF
1 BIRT
  2 DATE
  2 PLAC
  2 SOUR
1 DEAT
  2 DATE
```

```
    2 PLAC
    2 SOUR
```

Edit the template to create the new person's record. Change the name to the person's name. Assign the person's sex by deleting either **M** or **F**. Fill out the birth and death events as best you can. If the person is alive, remove the death event. Remove any DATE and PLAC lines you do not have the information for.

The default template provides lines for one birth and one death event. You can expand the record with other events (even more birth or death events) and lines. Indentation makes it easier to read and edit the record, but isn't necessary. You may change the default edit template by defining the user option INDIREC (described later).

Here is how I might edit the template when creating a record about myself:

Example 1.6 Example editing of template record

```
  0 INDI
  1 NAME Thomas Trask /Wetmore/ IV
  1 SEX M
  1 BIRT
    2 DATE 18 December 1949
    2 PLAC New London, New London, Connecticut
    2 SOUR Birth Certificate
  1 OCCU Software Engineer
  1 RESI
    2 DATE 1982 to 1995
    2 PLAC Newburyport, Essex, Massachusetts
    2 ADDR 2 Barton Street, Newburyport, MA 01950

... lots of other events and facts
```

When you edit a person record, don't add or modify INDI, FAMC or FAMS lines. LifeLines creates and maintains these lines through specific user commands.

When you finish editing and leave the editor, you automatically return to LifeLines. If you made an error (eg, didn't use proper level numbers or didn't follow the proper name convention), LifeLines displays an error message, and asks if you want to re-edit the record. If you don't, LifeLines doesn't add the person to the database.

When the record is in proper format, LifeLines asks if you are sure you want to add the person to the database. If you answer yes, the person is added; if you answer no, the person isn't. In both cases LifeLines returns to the main menu.

1.8 SCREEN EDITORS AND ENVIRONMENT VARIABLES

With LifeLines you maintain the database records using a screen editor. This is different than other genealogical programs where screens or forms are used to gather the data.The default screen editor for LifeLines is vi. (The MS-Windows version defaults instead to notepad.exe.) This can be overridden by the ED, EDITOR or LLEDITOR environment variables. For example, if you prefer the emacs screen editor, and if you use a bourne-compatible shell, you may add the line: **ED=*emacs*** to your login profile file, and LifeLines will use emacs for editing.

There are four other, LifeLines specific environment variables. They are LLDATABASES, LLARCHIVES, LLPROGRAMS and LLREPORTS. LLDATABASES and LLPROGRAMS are UNIX path list variables.

There is also a configuration file, and entries in it may be used in lieu of environment variables. It is ordinarily named .linesrc under UNIX, and lines.cfg under MS-Windows. A sample configuration file should have been included in the distribution.

See the section on System and User properties for more details.

LLDATABASES can be set to a list of directories that hold LifeLines databases. When you execute the LifeLines program, these directories will be searched in turn for the database mentioned on the command line. For example, **LLDATABASES=*.:/home/ttw4/Li***

indicates that databases should be searched for in the current directory first, and if not found there, then searched for in: `/home/ttw4/LifeLines/Databases`

Each LifeLines database is implemented as a directory with specific contents. The `LLDATABASES` variable should be set to a list of directories that contain these database directories, not to a list of database directories themselves.

The environment variable `LLPROGRAMS` is used in the same way, but to specify the search path for LifeLines report generating and other programs (described later).

`LLARCHIVES` and `LLREPORTS` can each be set to specify a single directory. `LLARCHIVES` is used to select a directory where all database backup files will be stored, and `LLREPORTS` is used to select a directory where all generated reports and program outputs will be placed.

New databases without explicit paths will be created in the first directory listed in the LLDATABASES path. (This is a change; versions from 3.0.6 to 3.0.31 used a now obsolete variable LLNEWDBDIR).

You are not required to use these environment variables; when a variable is not defined, LifeLines uses the current directory as its default value. If you do use the variables, you can override their use by specifying files and directories as either absolute or relative paths.

You may use the configuration file in lieu of environment variables. This is especially oriented towards users on MS-Windows systems, on which environment variables are not as common a configuration technique.

LifeLines uses the curses library for terminal independent I/O. This requires you to specify your terminal type with the `TERM` environment variable. (This is not relevant in the MS-Windows version.)

1.9 BASICS OF BROWSING

You will use the browsing screens of LifeLines most of the time. When in these modes you can quickly search for or browse through the persons and families in the database. When you find a person or family you are interested in, you can then edit their records.

The browsing screens also allow you to add new persons and families to the database, add spouses to families, add children to families, swap the order of spouses and children, merge persons and merge families, and perform other operations. The browsing screens also lets you remove spouses from families and remove children from families.

There are six browsing screens. The person and family screens concentrate on a single person and family, respectively. The list screen allows you to browse through a list of persons. The two person browse screen shows two persons at once, and the two family browse screen shows two families at once. The auxiliary screen is used browsing any other type of records (e.g., events, sources, notes).

Each browsing screen has multiple view modes. The view mode affects how the information is displayed on the screen, but does not affect the menu choices available at the bottom of the screen. Menu commands are available on each screen to change amongst the view modes available for that screen.

The person screen has the most view modes. It has normal mode, which shows a summary of the vital records of the person. It (like all other screens) has GEDCOM mode, which shows the actual GEDCOM data of the record, and also expanded GEDCOM mode, which shows the actual GEDCOM data, but augments it with information on each line that contains a cross-reference (GEDCOM xref). It has two pedigree or tree modes, one showing an ancestral tree and one showing a descendant tree. The depth of the pedigree trees shown may be adjusted via menu commands.

The two person browse screen has the same modes as the person screen.

The two family browse screen and tandem family screen alike have normal mode (showing a summary of vitals), GEDCOM mode, and expanded GEDCOM mode.

The auxiliary screen has only GEDCOM mode and expanded GEDCOM mode. (The list screen has no view modes at present).

1.10 IDENTIFYING A PERSON OR LIST OF PERSONS TO BROWSE

To enter the browsing modes from the main menu strike b. LifeLines asks you to identify a person or list of persons to browse to:

```
Please identify person or persons to browse to.
Enter name, key, refn or list:
```

Enter either a name or partial name, or an internal key value, or a user-defined reference key (described later) or the name of a previously defined list of persons (described later), and strike return.

LifeLines allows wide flexibility in how to enter names. You may enter a name in upper or lower case or any combination. You may leave out all but the first given name, and, for given names, you may leave out any letters except the first. You may leave vowels out of the surname, and after four or five consonants have been typed, you may leave them out too. You must separate the given names from the surname by a slash, and if you enter given names after the surname (as in Chinese names), or any modifiers (such as Jr, Sr, IV, etc.), they must be separated from the surname by another slash. Here are a few of the ways I can enter my name:

Example 1.7 Example of entering a name

```
Thomas Trask /Wetmore/ IV
thomas/wetmore/iv
t t/wetmr/i
th tr/Wetmore
t/wtmr/iv
```

You may browse to the list of all persons with the same surname by using the * character as the first initial. For example:

Example 1.8 Example of using wildcard in browsing

```
*/wetmore
```

matches all persons with surname Wetmore. This is the only wildcard feature supported in browsing. (However, the search operation provides some simple wildcards for finding individual name fragments, or searching by user-defined reference keys. The search operation is accessed via a different choice off of the main menu.)

After you enter a name, LifeLines searches for all persons who match. There are three possibilities: no one matches; one person matches; or more than one person matches. In the first case LifeLines writes:

```
There is no one in the database with that name or key.
```

and leaves you in the main menu.

If one person matches, LifeLines enters the person browse mode displaying the matched person. If more than one person matches, LifeLines enters the list browsing mode with the list of matching persons.

You may also identify a person by entering his or her internal, cross-reference key value. The internal key values of all person records are an I followed by digits. When you enter a key value you may omit the I. If LifeLines finds a person with the key value you provide, LifeLines enters the person browsing mode displaying that person. You can also browse to a Family, Source, or Note by entering its key, but you must include the letter identifing the key type, thus F11, S1, or N3 would browse to the family, source or note corresponding to the key if it exists.

The browse command b is also available from most browsing modes. The command works the same way from those modes as it does from the main menu.

1.11 ZIP IDENTIFYING A NEW PERSON

Some LifeLines operations need you to identify a person, not for the purpose of browsing, but for the purpose of completing an operation you have requested. For example, when you add a child to a family, LifeLines may ask you to identify the child. When this happens a panel pops up that asks you to identify a person. You respond by typing a name or key exactly as you would for the b command. If no one matches, LifeLines returns to the previous browsing mode. If the name matches persons in the database LifeLines displays something like:

```
Please choose from among these records.
  >Thomas Trask Wetmore, b. 1826, N.B. (42)
   Thomas Trask Wetmore IV, b. 1949, Conn. (1)
   Thomas Trask Wetmore III, b. 1925, Conn. (6)
   Thomas Trask Wetmore Jr, b. 1896, Conn. (11)
   Thomas Trask Wetmore Sr, b. 1866, N.S. (23)
   Thomas Trask Wetmore V, b. 1982, Mass. (5)
_____________________________________________
Commands: j Move down k Move up i Select q Quit
```

Use the j and k commands to move the selection cursor (>) to the correct person, and then use the i command to select that person. There may be more persons in the list than you can see at once. If this is so then you can use the j and k commands to scroll through the full list. If you don't find the proper person, use the q command and LifeLines asks whether you want to enter another name.

With version 3.0.15, lists may also be navigated with the up and down arrows, PageUp and PageDown keys, Home and End keys, and the Enter key. Shift-PageUp and Shift-PageDown move more than one page at a time in a given direction. The keyboard equivalents are j=UpArrow, k=DownArrow, u=PageUp, d=PageDown, ^=Home, $=End, U=Shift-PageUp, D=Shift-PageDown, i=Enter.

When LifeLines creates a list of names for you to select from, it tries to add extra information to the name; this helps determine which name to choose, and is important in databases where many persons have the same name. LifeLines also places the person's key value at the end of each menu line; this may be helpful in large databases.

Some browse screens provide the z command, which allows you to browse to a new person using the zip style of identification rather than the b style.

1.12 BROWSE DISPLAY BASICS

The screen display for each browsing screen is made up of panels. At the bottom of each display is a message panel used for one line messages. Each display contains one or two data panels showing information from the database. And each display has a panel with the operation menu available for that screen.

1.13 PERSON BROWSE SCREEN

After you identify a person to browse to, LifeLines enters the person browse screen. The top panel in the display gives basic information about the person (in the normal, or vitals, mode, which is the default). The middle panel provides a menu of commands. For example:

```
person: Thomas Trask WETMORE Sr (25)
  born: 13 March 1866, St. Mary's Bay, Digby, Nova Scotia
  died: 17 February 1947, New London, New London, Connecticut
  father: Daniel Lorenzo WETMORE, b. 1821, N.S., d. 1903, Conn. (48)
  mother: Mary Ann DOTY, b. 1824, N.S., d. 1897, Conn. (59)
  spouse: Margaret Ellen KANEEN, b. 1855, Eng., d. 1900, Conn. (26)
    child: Portia Louise WETMORE, b. 1892, Conn., d. 1921, Conn. (27)
    child: Thomas Trask WETMORE, b. 1896, Conn., d. 1970, Conn. (17)
  spouse: Arleen M KEENEY, m. 1914, Conn. (75)
_____________________________________________________________________
Please choose an operation:             (pg 1/3)
  e  Edit the person       g  Browse to family    p  Pedigree mode
  f  Browse to father      u  Browse to parents   n  Create new person
  m  Browse to mother      b  Browse to persons   a  Create new family
  s  Browse to spouse/s    h  Add as spouse       x  Swap two families
  c  Browse to children    i  Add as child        tt Enter tandem mode
  o  Browse to older sib   r  Remove as spouse    ?  Other menu choices
  y  Browse to younger sib d  Remove as child     q  Return to main menu
```

```
LifeLines -- Person Browse Screen
```

The commands perform a wide variety of functions.

e Edit the person.

Edit the person's database record. LifeLines puts the record in a file, and then runs a screen editor so you can edit the record. When you return from the editor, LifeLines asks you to confirm any changes; the person is changed only if you answer yes.

f Browse to father.

Browse to the person's father. If the father isn't in the database, LifeLines doesn't change the display. If there are more than one father, LifeLines asks you to select one.

m Browse to mother.

Browse to the person's mother. If the mother isn't in the database, LifeLines doesn't change the display. If there are more than one mother, LifeLines asks you to select one.

s Browse to spouse/s.

Browse to the person's spouse. If the person has more than one spouse, LifeLines asks you to select one. If the person has no spouse, the display does not change.

c Browse to children.

Browse to one of the person's children. If there is more than one child, LifeLines asks you to select one. If the person has no children, the display does not change.

o Browse to older sib.

Browse to the person's next older sibling. If the person has no such sibling, the display does not change. Only siblings from the same family are browsed by this command.

y Browse to younger sib.

Browse to the person's next younger sibling. If the person has no such sibling, the display does not change. Only siblings from the same family are browsed by this command.

g Browse to family.

Browse to the family the person is a spouse or parent in, and switch to the family browse mode. If the person is in more than one family, LifeLines asks you to identify which one. If the person is not a spouse or parent in any family, the display does not change.

u Browse to parents.

Browse to the family the person is a child in, and switch to the family browse mode. If the person is not a child in a family, the display does not change. If the person is a child in more than one family, LifeLines asks you to identify which one.

b Browse to persons.

Browse to a new person or list of persons. LifeLines asks you to identify a person or persons by name, key or list name, and depending on how many persons are identified, switches either to the list browse mode, or remains in the person browse mode.

h Add as spouse.

Add the person as a spouse/parent to an existing family. LifeLines asks you to identify the family, and then asks you to confirm the request.

i Add as child.

Add the person as a child to an existing family. The person may already be a child in another family. LifeLines asks you to identify the family, and then asks you to confirm the request.

r Remove as spouse.

Remove the person as a spouse or parent from an existing family. If the person is a spouse or parent in more than one family, LifeLines asks you to identify the family.

d Remove as child.

Remove the person as a child in an existing family.

n Create new person.

Create and add a new person to the database. LifeLines creates a record template and puts you into the screen editor to edit the record. When you return from the editor, LifeLines asks you to confirm the operation. If you do, the new person is added and becomes the current person. If not, the new person is not added, and LifeLines returns to the original display.

a Create new family.

Create and add a new family to the database. The new family may have the current person as either a spouse/parent or as a child; LifeLines asks which. If you choose to create a family with the person as a spouse/parent, LifeLines asks you to identify the other spouse if he or she is known. In either case LifeLines creates a family template, and places you in the screen editor. When you return from the editor, LifeLines asks you to confirm the operation. If you do, LifeLines adds the family and shifts into family browse mode. If the command you ran just before the a command were the n command, and you choose to create a family with the person as a spouse/parent, LifeLines guesses that the other spouse in the family will be the person displayed just before the new person was created. LifeLines asks you if this is the case, and if so, automatically make that person the other spouse in the new family. If this is not the case, LifeLines asks you to identify the other spouse.

p Pedigree mode.

Change to pedigree browse mode. The person becomes the root person in the pedigree display.

x Swap two families.

Swap (change chronological order) any two families that the person belongs to as a spouse or parent. LifeLines asks you to identify the two families and then swaps them.

tt Enter tandem mode.

Change to the tandem person browse mode. LifeLines first asks you to identify the second person.

? Other menu choices.

Go to the next page of menu choices. This will have no effect on the upper (data) portion of the screen, but it allows you to page through all available commands for this display screen.

q Return to main menu.

Return to the LifeLines main menu.

z Zip browse to person.

Zip browse to a new person. LifeLines asks you to identify a person by name or key value, and if you do so, browses to that person.

1.14 LIST BROWSE SCREEN

This browse screen handles lists of persons. The top panel shows information about one person in the list. The left panel shows a list of up to 12 persons. The person shown in the top panel is identified by the > character. The right panel is the menu of available commands.

```
person: Thomas Trask WETMORE Sr (25)
  born: 13 March 1866, St. Mary's Bay, Digby, Nova Scotia
  died: 17 February 1947, New London, New London, Connecticut
  father: Daniel Lorenzo WETMORE, b. 1821, N.S., d. 1903, Conn. (48)
  mother: Mary Ann DOTY, b. 1824, N.S., d. 1897, Conn. (59)
  spouse: Margaret Ellen KANEEN, b. 1855, Eng., d. 1900, Conn. (26)
```

```
  Thomas Trask WETMORE (42)                Choose an operation:
  Thomas Trask WETMORE III (6)              j Move down list
  Thomas Trask WETMORE IV (1)               k Move up list
  Thomas Trask WETMORE (11)                 e Edit this person
 >Thomas Trask WETMORE Sr (23)              i Browse this person
  Thomas Trask WETMORE (5)                  m Mark this person
                                            r Remove from list
                                            t Enter tandem mode
                                            n Name this list
                                            b Browse new persons
                                            a Add to this list
                                            x Swap mark/current
                                            q Return to main menu
LifeLines -- List Browse Screen
```

j Move down list.

Move down the list one person. The list panel is only large enough to show 12 persons. However, the list may contain many more persons. Use the **j** and **k** commands to scroll to these other persons.

k Move up list.

Move up the list one person. The list panel is only large enough to show 12 persons. However, the list may contain many more persons. Use the **j** and **k** commands to scroll to these other persons.

With version 3.0.15, lists may also be navigated with the up and down arrows, PageUp and PageDown keys, Home and End keys, and the Enter key. Shift-PageUp and Shift-PageDown move more than one page at a time in a given direction. The keyboard equivalents are j=UpArrow, k=DownArrow, u=PageUp, d=PageDown, ^=Home, $=End, U=Shift-PageUp, D=Shift-PageDown, i=Enter.

e Edit this person.

Edit the displayed person's database record. LifeLines runs the editor on the person's record. When you return from the editor, LifeLines asks you to confirm any changes, and then leaves you in the list browse screen.

i Browse this person.

Change to the person browse screen with the current person.

m Mark this person.

Mark the current person if he/she is not marked; unmark the person is he/she is. The marked person is shown with an x by his/her name. Marked persons are used by the t and x commands. Only one person may be marked at a time.

r Remove from list.

Remove the current person from the browse list (not from the database).

t Enter tandem mode.

Change to the tandem person mode with the current person and the marked person as the two persons. If no person is marked there is no change.

n Name this list.

Lists of persons may be named, allowing you to quickly browse back to them by giving a list name in response to the b command from different modes. LifeLines will prompt you for the name. List names are most convenient when short.

b Browse new persons.

Browse to a new person or list of persons. You can identify a person or list of persons by name, internal or user key or by list name.

a Add to this list.

Add more persons to the current browse list. LifeLines asks you to identify a new person or list of persons by name, key or list name, and they are added to and name-sorted into the current list.

x Swap mark/current.

Swap the current person with the marked person in the list.

q Return to main menu.

Return to the LifeLines main menu.

1.15 FAMILY BROWSE SCREEN

This browse screen displays information about a family. The top panel shows basic information about the family. The bottom panel shows the menu of available commands. If the database contains more than two parents for this family only the first two are displayed.

```
father: Thomas Trask WETMORE IV (1)
  born: 18 December 1949, New London, New London, Connecticut
  died:
mother: Luann Frances GRENDA (2)
  born: 10 July 1949, Pittsburgh, Allegheny, Pennsylvania
  died:
married: 1 August 1970, Governors Island, New York, New York
  child: Anna Vivian Wetmore, b. 1974, Alaska (3)
  child: Marie Margaret WETMORE, b. 1979, Conn. (4)
  child: Thomas Trask WETMORE V, b. 1982, Mass. (5)
_____________________________________________________________________________
Please choose an operation:            (pg 1/4)
 e  Edit the family       %s  Add source           r  Remove spouse from
 f  Browse to father      %e  Add event            d  Remove child from
 m  Browse to mother      %o  Add other            x  Swap two children
 c  Browse to children    s  Add spouse to family  ?  Other menu choices
 n  Create new person     a  Add child to family   q  Return to main menu
_____________________________________________________________________________
LifeLines -- Family Browse Screen (* toggles menu)
```

e Edit the family.

Edit the family's record. LifeLines writes the record to a file and puts you into an editor to edit the file. When you return from the editor, LifeLines asks you to confirm the update; the family is changed only if you do so.

f Browse to father.

Browse to the father/husband of the family, switching to person browse screen. If the father is not there, there is no change.

mBrowse to mother.

Browse to the mother/wife of the family, switching to person browse screen. If the mother is not there, there is no change.

c Browse to children.

Browse to a child in the family, switching to the person browse screen. If the family has more than one child, LifeLines asks you to identify a specific child.

n Create new person.

Create and add a new person to the database. LifeLines creates a record template and puts you into the screen editor to edit the record. When you return from the editor, LifeLines asks you to confirm the operation. If you do, the new person is added to the database. If not, the new person is not added. In both cases the display does not change.

s Add spouse to family.

Add a spouse to the family. LifeLines asks you to identify the new spouse. If the command you ran just before the s command were the n command, LifeLines guesses that the new spouse will be the person just created. LifeLines asks if this is the case, and if so, makes that person the second spouse in the family. If not, LifeLines asks you to identify the other spouse.

a Add child to family.

Add a child to the family. LifeLines asks you to identify the new child. If the command you ran just before the a command were the n command, LifeLines guesses that the new child will be the person just created. LifeLines asks if this is the case, and if so, adds that child to the family. If not, LifeLines asks you to identify the child. If the family already has children, LifeLines also asks where to place the new child in the family.

r Remove spouse from.

Remove a parent/spouse from the family. LifeLines asks you to identify the person, and if you do, removes him or her. The person is not removed from the database.

d Remove child from.

Remove a child from the family. LifeLines asks you to identify the child should, and if you do, removes the child from the family. The person is not removed from the database.

x Swap two children.

Swap (change the chronological order) of any two children in the family. LifeLines asks you to identify the two children and then swaps them.

t Enter family tandem.

This command takes you to the tandem family browse screen. LifeLines asks you to identify a second family, and then takes you to the tandem family screen, displaying both the two families.

b Browse to persons.

Browse to a new person or list of persons. You can identify a person or list by name, by key, or by list name. If you successfully identify a new person or persons you will switch into the person or list browse screens.

z Browse to person.

Zip browse to a new person. LifeLines asks you to identify a person by name or key value, and if you do, browses to that person.

q Return to main menu.

Return to the LifeLines main menu.

1.16 TANDEM PERSON BROWSE MODE

The tandem person browse screen displays information about two persons. Its main use it to support the person merging operation. The top two panels show two persons in the format used in the person and list screen displays. The bottom panel gives the menu of available commands. For example:

```
  person: Thomas Trask WETMORE Sr (25)
  born: 13 March 1866, St. Mary's Bay, Digby, Nova Scotia
  died: 17 February 1947, New London, New London, Connecticut
  father: Daniel Lorenzo WETMORE, b. 1821, N.S., d. 1903, Conn. (48)
  mother: Mary Ann DOTY, b. 1824, N.S., d. 1897, Conn. (59)
  spouse: Margaret Ellen KANEEN, b. 1855, Eng., d. 1900, Conn. (26)
```

```
person: Thomas Trask WETMORE IV (1)
  born: 18 December 1949, New London, New London, Connecticut
  died:
  father: Thomas Trask WETMORE III, b. 1925, Conn. (6)
```

```
  mother: Joan Marie HANCOCK, b. 1928, Conn. (7)
  spouse: Luann Frances GRENDA, m. 1970, N.Y. (2)
_____________________________________________________________________
Please choose an operation:
 e Edit top person    s Browse top spouse/s   a Add family
 t Browse to top      c Browse top children   j Merge bottom to top
 f Browse top father  b Browse to persons     x Switch top/bottom
 m Browse top mother  d Copy top to bottom    q Return to main menu
_____________________________________________________________________
LifeLines - Two Person Browse Screen
```

e Edit top person.

Edit the top person's record. LifeLines writes the record to a file, and puts you in the screen editor to edit the file. When you return from the editor, LifeLines asks you to confirm the update; the person is changed only if you do so.

t Browse to top.

Switch to the person display with the top person as current person.

f Browse top father.

Replace the top person with his/her father.

m Browse top mother.

Replace the top person with his/her mother.

s Browse top spouse/s.

Replace the top person with his/her spouse. If the person has more than one spouse, LifeLines asks you to identify one.

c Browse top children.

Replace the top person with one of his/her children. If the person has more than one child, LifeLines asks you to identify the one.

b Browse to persons.

Browse to a new person or list of persons. LifeLines asks you to identify a new person or persons by name, key or list name, and then does as described in the section on identifying a person.

d Copy top to bottom.

Copy the top person into the bottom person. A new person is not created; the same person is displayed twice.

a Add family.

Create a new family record; LifeLines assumes the two displayed persons are to become the spouses/parents in the new family; they must be of opposite sex.

j Merge bottom to top.

Merge the bottom person into the top person. LifeLines combines the two person records and places you in the screen editor to edit the combined record. When you are done, if you confirm the operation, LifeLines removes the bottom person from the database, and the top person is given the combined record. See the section on merging.

x Switch top/bottom.

Swap the two persons in the display.

q Return to main menu.

Return to the LifeLines main menu.

1.17 TANDEM FAMILY BROWSE MODE

The tandem family browse screen displays information about two families. Its main use it to support the family merging operation.The top two panels provide information about the two families you are browsing, and the bottom panel holds the menu of available commands. For example:

```
father: Thomas Trask WETMORE IV (1)
  born: 18 December 1949, New London, New London, Connecticut
mother: Luann Frances GRENDA (2)
  born: 10 July 1949, Pittsburgh, Allegheny, Pennsylvania
married: 1 August 1970, Governors Island, New York, New York
  child: Anna Vivian WETMORE, b. 1974, Alaska (3)
_____________________________________________________________________
father: Thomas Trask WETMORE III (6)
  born: 26 October 1925, New London, New London, Connecticut
wife: Joan Marie Hancock (7)
  born: 6 June 1928, New London, New London, Connecticut
married: 5 February 1949, New London, New London, Connecticut
  child: Thomas Trask WETMORE IV, b. 1949, Conn. (1)
_____________________________________________________________________
Please choose an operation:            (pg 1/3)
 e  Edit top person       m  Browse to mothers     )b Scroll bottom down
 t  Browse to top         (t Scroll top up         (( Scroll both up
 b  Browse to bottom      )t Scroll top down       ?  Other menu choices
 f  Browse to fathers     (b Scroll bottom up      q  Return to main menu
_____________________________________________________________________
LifeLines -- Two Family Browse Screen (* toggles menu)
```

e Edit top family.

This command lets you edit the top family's record. LifeLines writes the record into a file, and then puts you into an editor to edit that information. When you return from the editor, LifeLines asks you whether you are sure you want to update the family in the database. The family is changed only if you answer yes.

t Browse to top.

Change to the family browse screen with the top family the current family.

b Browse to bottom.

Change to the single family browse screen with the bottom family the current family.

f Browse to fathers.

Change to the tandem person screen with the fathers of the two families as the two persons.

m Browse to mothers.

Change to the tandem person screen with the mothers of the two families as the two persons.

(t Scroll top up.

TODO

)t Scroll top down.

TODO

(b Scroll bottom up.

TODO

)b Scroll bottom down.

TODO

((Scroll both up.

TODO

)) Scroll both down.

TODO

 # Toggle childnos.

TODO

(1-9) Browse to child.

TODO

j Merge bottom to top.

Merge the bottom family into the top family. LifeLines combines the two family records and places you in the screen editor to edit the combined record. When you are done, if you confirm the operation, LifeLines deletes the bottom family from the database, and the top family is given the combined record. See the section on merging.

x Switch top/bottom.

Swap the two families in the display.

[There are some more miscellaneous commands available on the menus.]

q Return to main menu.

Return to the LifeLines main menu.

1.18 PEDIGREE BROWSE MODE

The pedigree browse screen displays a four-generation pedigree for the current person. The top panel holds the pedigree, and the bottom panel holds the menu of available commands. For example:

```
                  John WETMORE [1755-1848] (32)
            Daniel Van Cott WETMORE [1791-1881] (41)
                  Anna VAN COTT [1757-1802] (33)
      Daniel Lorenzo WETMORE [1821-1903] (48)
                  Thomas TRASK [-1836] (81)
            Hannah TRASK [1797-1829] (46)
                  Susannah PORTER [1754-] (82)
Thomas Trask WETMORE Sr [1866-1947] (25)
                  Samuel DOTY [1759-] (501)
            Samuel DOTY [1787-] (74)
                  Hephzibah PORTER [1764-1853] (502)
      Mary Ann DOTY [1827-1897] (59)
                  Nathan SAVERY [1748-1826] (510)
            Lydia SAVERY [1806-] (75)
                  Deidamia SABEAN [1765-1845] (511)
_____________________________________________________________________________
Please choose an operation:
 e Edit the person    m Browse to mother     g Browse to family
 i Browse to person   s Browse to spouse/s   b Browse to persons
 f Browse to father   c Browse to children   q Return to main menu
_____________________________________________________________________________
LifeLines - Pedigree Browse Mode
```

e Edit the person.

Edit the current person.

i Browse to person.

Change to the person display mode with the current person.

f Browse to father.

Browse to the father of the current person, shifting the pedigree one generation back. If the father is not in the database, there is no change.

m Browse to mother.

Browse to the mother of the current person, shifting the pedigree one generation back. If the mother is not in the database, there is no change.

s Browse to spouse/s.

Browse to a spouse of the current person, shifting the display to the pedigree of that person. If the current person has more than one spouse, LifeLines asks you to identify the spouse; if the person has no spouse there is no change.

c Browse to children.

Browse to a child of the current person, shifting the pedigree one generation forward. If the current person has more than one child, LifeLines asks you to identify the child; if the person has no children there is no change.

g Browse to family.

Change to the family display; the family will be the one that the current person belongs to as spouse or parent. If there are more than one, LifeLines asks you to identify the proper one.

b Browse to persons.

Browse to another person or list of persons; if you identify a single person the display remains in the pedigree display; if you identify more than one person the display changes to the list browse mode.

q Return to main menu.

Leave the pedigree browsing mode and return to the main menu.

1.19 SEARCH MENU

If you choose Search database from the main menu, LifeLines displays the search menu:

```
How would you like to find a record?
  v Review visit history (12 records)
  c Review change history (3 records)
  f Full database scan
  q Return to previous menu
```

The first two items will depend on your previous activity. If you have browsed to individuals or family records in the database, the first item will appear similar to what's shown above, if you haven't it will just contain a note that the visit history is empty. The second item will appear similar to what's shown above if you have changed individual records in this session with LifeLines, otherwise it will contain a note that the change history is empty.

Selecting a non-empty visit history or change history will bring up a list of individuals (or families) that are in the history, allowing you to browse to that individual or family.

If you choose Full database scan off the search menu, LifeLines displays the fullscan menu.

```
What scan type?
  f  Full name scan
  n  Name fragment (whitespace-delimited) scan
  r  Refn scan
  q  Return to previous menu
```

The first two items on this menu allow you to search all the NAME records in the current database. If you choose Full name scan you are prompted for a search pattern and then LifeLines searches for all the individual NAME records whose value matches the pattern supplied. If you choose the Name fragment scan, you will be prompted for a search pattern and then LifeLines will search for whitespace delimited words within individual NAME records that match the pattern supplied.

1.19.1 Search Patterns

The pattern supplied to the search commands is used to match against the names in the database. The following characters have special meaning when used in a pattern:

?	Matches any single character
*	Matches zero or more occurrences of any character
SUB	a control Z (^Z) is similar to '*', this matches zero or more occurrences of any characters other than '.'. (of course you probably can't type this on unix)
[ab]	A set of characters enclosed in square brackets matches the single characters listed between the brackets. If the dash '-' character is to be included, it must immediately follow the opening bracket '['. If the closing bracket ']' character is to be included, it must be preceded by a quote '`'.
[a-z]	Matches a single character in the range 'a' to 'z'. Ranges and sets may be combined within the same set of brackets.
!R	Matches a single character not in the range 'R'. If range 'R' includes the dash '-' character, the dash must immediately follow the '!'.
!	Makes the following pattern match any string except those what it would normally match.
QUOTE	(On DOS this is `, on UNIX it is \) Makes the next character a regular (nonspecial) character. Note that to match the quote character itself, it must be quoted. Note that this character must be escaped if used within string constants ("\\").

Upper and lower case alphabetic characters are considered identical, i.e., 'a' and 'A' match each other. (What constitutes a lowercase letter depends on the current locale settings.)

Spaces and control characters (other than control z) are treated as normal characters,

As an example, consider the following NAME record:

```
1 NAME John /Smith/
```

When doing a full name search, the value searched is the complete gedcom name value including the slashes used to delimit the surname. in order to have a match, the search pattern must match the slashes. Thus

```
*smith
```

will not match this name, whereas

```
*smith/
```

will match.

When doing a name fragment search, the slashes are removed from the surname before trying to match the name, thus

```
smith
smi*
joh*
*hn
```

will all match this NAME record.

1.20 ADD OPERATION

If you choose the add operation from the main menu, LifeLines displays the add menu:

```
What do you want to add?
  p Person - add new person to the database
  f Family - create family record from one or two spouses
  c Child - add a child to an existing family
  s Spouse - add a spouse to an existing family
  q Quit - return to the previous menu
```

These operations work in a straightforward way. LifeLines asks you the necessary questions, and lets you cancel at any time. The operations provided by this menu are also available from the browsing modes, and are often easier to perform there.

1.21 DELETE OPERATION

If you choose the delete operation at the main menu, LifeLines displays the delete menu:

```
What do you want to delete?
  c Child - remove a child from his/her family
  s Spouse - remove a spouse from a family
  p Person - remove a person completely
  q Quit - return to the previous menu
```

These operations also work in a straightforward way. LifeLines asks you the necessary questions and lets you cancel at any time.

You may also remove a child from his/her family, or remove a spouse/parent from his/her family, from the person browsing mode. In both cases, only a relationship is removed, not a person. On the other hand, the delete menu must be used if you want to completely remove a person from the database; this cannot be done from the browsing mode.

There is no special operation for removing a family record. LifeLines silently removes any family record that has no parent or child associated with it.

1.22 CODESET CONVERSION

(This section was previously entitled CHARACTER TRANSLATION.)

The intention is that you need only specify the internal codeset for each database you create (and this step may be automated via the NewDbProps property), and all else works pretty well without tuning. That is, lifelines tries to guess the correct codeset for your environment (including guessing the console and windows codesets when operating under MS-Windows, which it should do fairly well).

However, you may encounter situations where you wish to alter the codeset behavior, or the codeset conversion is not operating correctly (in which case we hope you will report the problem to the mailing list and/or github bugs list).

There are two ways to amend codeset conversion. The first method is by changing configuration variables. For example, if you wish to generate an HTML report of all your data, which includes names in Russian (in Cyrillic letters), for your cousin, and you know that your cousin's computer has no font for Cyrillic letters, you might wish to temporarily adjust your report output codeset so that you will get interpolated ASCII letters for the Russian letters. You could do this by temporarily altering the configuration variable ReportCodesetOut to be "ASCII" (actually, if any of your data has characters in it that are reserved in HTML, such as the less than sign, or the ampersand, you would probably want "ASCII//HTML").

The second way to change codeset conversion, and the only way in lifelines 3.0.6, is to edit the embedded character translation tables, in which you actually specify the letters you want converted, letter by letter, and how you want them converted. This method, unlike the first, even works in databases with no specified internal codeset.

If you choose the modify character translation tables operation from the main menu, LifeLines displays the character translation menu:

```
Which character mapping do you want to edit?
   e Editor to Internal mapping
   m Internal to Editor mapping
   i GEDCOM to Internal mapping
   x Internal to GEDCOM mapping
   d Internal to Display mapping
   r Internal to Report mapping
   q Return to main menu
```

LifeLines can do codeset conversion in changing text from one form to another, and lifelines supports five different forms.

internal

for records in the database

editor

for records being edited

display

for records being displayed

report

for records written to output file

GEDCOM

for records read in from or written out to GEDCOM

When converting text from one form to another LifeLines normally uses iconv conversion, and codesets specified in configuration variables. This may be augmented by codeset translation or extension using the text conversion (*.tt) files in the tt subdirectory. To use the tables in the tt subdirectory, you need to set the property "TTPATH" in your LifeLines configuration file to the path of the tt directory. There are two types of files in this directory.

Files of the form <codeset>_<codeset1>.tt convert from one codeset to another. For example, CP1250_UTF-8.tt can be used to convert characters in codeset CP1250 to their representations in UTF-8.

Files of the form <codeset>__<subcodeset>.tt apply a conversion within the codeset, for example, UTF-8__html.tt is a sub-conversion that converts UTF-8 characters that have special escape codes within html to those special codes. For example, specifying the report codeset to be UTF-8//html will apply the html sub-conversion to all the data being written. Probably not what you really wanted. See the report language function convertcode() in the reportmanual for details.

If your system lacks iconv, or you need more specialized conversion than provided with iconv, you may either write a text conversion file (a tt file), or you may edit one of the in-database translation tables.

The in-database translation tables convert between forms (as listed above). Every translation table converts either to the internal form, or from the internal form. That is, the internal form is used as an intermediate step in all operations. There are six supported translation tables. The following table shows the six tables and describes when they are applied:

internal to editor

when converting from internal, database form to editor form

editor to internal

when converting from editor form back to internal, database form

GEDCOM to internal

when reading GEDCOM input records and writing them to database

internal to GEDCOM

when writing internal database records to external GEDCOM file

internal to display

when displaying a record in a browsing mode display screen

internal to report

when writing internal database records to external report file

After you select a translation table you are placed in the editor to edit the table. Translation tables are made up of lines that look like:

```
pattern pattern
```

where a tab separates the patterns. Each pattern is an arbitrary sequence of verbatim ASCII characters and escape sequences. Translation occurs by finding all occurrences that match left patterns and replacing them with the corresponding right patterns.

There are five escape mechanisms used in patterns:

#nnn

nnn is a decimal character value

$hh

hh is a hexadecimal character value

\ #

represents the # character

\$

represents the $ character

****

represents the \ character

It is possible, and desirable, to provide a short name for the translation table, using the "##!name: " command. An example would be

```
##!name: UTF-8 to latex
```

Naming the translation table is desirable because these names are displayed, at least in part and if they fit, on the translation table menu.

It is possible to format the file using a character other than tab as the separator between source and destination code. To do requires using the "##!sep" command. Those exact six characters must begin the line, and then the next character is the new separator for all following lines. For clarity, this should only occur once, and near the top of the file before any actual translation lines, and a fairly clear separator should be used (e.g., the equal sign "=").

Any line which is blank, or which begins with two hash marks (##), is ignored. Therefore, comments begin with two hash marks.

For advanced users, it is possible to mix different types of conversion, for example iconv conversion and also translation table conversion, in the same form step. For example, it is possible to convert internal database text (internal form) first via the "internal to GEDCOM" in-database translation table, and then via the iconv conversion from configured internal codeset to configured GEDCOM codeset. In-database translation tables are always applied in the internal codeset, so when converting to the internal form, they are applied after iconv and/or tt conversions, and when converting from internal form, they are applied first.

An example of adding a mixin in-database translation table might be to escape certain characters which are control characters to an output computer language, e.g., latex. One could create an "Internal to Report" mapping in UTF-8 (if the database is internally UTF-8) to escape any characters that may occur in place names or textual descriptions and inadvertently cause grief in latex processing.

However, in this case, one could also write a tt file to achieve the same results, and be shared across databases, by naming it, eg, UTF-8__latex.tt. The double underscore ("__") signifies that this is a conversion to be applied to text which is in UTF-8, and to trigger Lifelines to use this, one must specify a report codeset such as "UTF-8//latex" (if UTF-8 output is desired, but with the latex conversion first applied), or "ISO-8859-1//latex" (if ISO-8859-1 output is desired, but with the latex conversion first applied).

1.23 MISCELLANEOUS UTILITIES

If you choose the miscellaneous utilities operation, LifeLines displays the utilities menu:

```
What utility do you want to perform?
  s Save the database in a GEDCOM file
  r Read in data from a GEDCOM file
  R  Pick a GEDCOM file and read in
  k Find a person's key value
  i Identify a person from key value
  d Show database statistics
  m Show memory statistics
  e Edit the place abbreviation file
  o Edit the user options file
  c  Character set options
  q Return to the main menu
```

s Save the database in a GEDCOM file.

This command saves the complete LifeLines database in a GEDCOM file. All person, family, event, source and user-defined records are stored. This command may be used to periodically back up your database. When you use this command, LifeLines asks you for the name of the file. If you have defined the `LLARCHIVE` shell variable, LifeLines will store the file in the directory named in the variable.

r Read in data from a GEDCOM file.

This command allows you restore a complete database from a GEDCOM file. When you select this command, LifeLines asks you for the name of the GEDCOM file. This command can also be used to import data from a GEDCOM file to an existing database. When LifeLines performs this command, it first reads the entire GEDCOM file and checks it for validity. If there are problems in the file, LifeLines describes them, writing them to the file err.log, and does not add any records to the database. If there are no problems, LifeLines adds all the records found in the file to the database (only header and trailer records are not stored in the database).

Normally, LifeLines will replace the XREF's (the identifiers for individuals, families, sources, notes, etc.) in the GEDCOM with its own. These identifiers are reserved by the GEDCOM for internal use of genealogy programs. However, if the identifiers are compatible with LifeLines internal representation and there are no conflicts with existing identifiers, LifeLines will ask you if you want to preserve the identifiers in the GEDCOM file rather than assign new values.

k Find a person's key value.

This command finds the internal key value of a person.

i Identify a person from key value.

This command identifies the person that has a particular internal key value.

d Show database statistics.

This command summarizes the contents of the current database. It displays the number of person, family, source, event and other records in the database.

m Show memory statistics.

This command is used by the author for debugging.

e Edit the place abbreviation file.

This command allows you to edit the place abbreviations file. This file defines the abbreviations that are used by LifeLines when it creates lists of persons for you to select from. Each line in the file has the format:

```
word:
      abbr
```

where word is a word to be abbreviated, and abbr is its abbreviation. The word and its abbreviation are separated by a colon. For example:

Example 1.9 Example of using abbreviations

```
Connecticut:Conn.
Massachusetts:Mass.
Nova Scotia:N.S.
```

When LifeLines constructs lists of persons for you to select from, it looks up the last component of certain `PLAC` lines in this file, and if it finds that component, replaces it with its abbreviation.

o Edit the user options file.

This command allows you to edit the user options file. The user options file is a record kept in the database that holds user selectable options. Each option has a name and a string value. Each line in the options file has the format:

```
option=value
```

where option is the name of an option and value is the option's string value. If the value is more than one line long, then the last character in each non-final line must be a backslash. The escapes \n and \t are also recognized in version 3.0.7 and better (to represent a carriage return and a tab, respectively). These (\n and \t) are primarily for use in custom record templates. In version 3.0.14 there are 39 options. Note that all of these may also be specified in the configuration file, to apply to all databases, but if specified in both places, the entry in the database (user options table) governs. For more information, see the sample configuration file; each option is preceded by a brief explanation.

The list of options can be found in the section 'System and User Properties' below.

For example if you would like to replace the default person record template with the following:

Example 1.10 Example of replacing the default person record template

```
0 INDI
1 NAME //
1 SEX
```

you would edit the user option file to contain:

Example 1.11 Example of replaced default person record template

```
INDIREC=0 INDI\
  1 NAME //\
  1 SEX
```

or, using the \n escape so as to keep the entry on one line:

Example 1.12 Example of replaced single line default person record template

```
INDIREC=0 INDI\n1 NAME //\n1 SEX
```

q Return to main menu.

This command returns you to the main menu.

1.24 IMPORT ERRORS

Errors generated during a GEDCOM import are logged to a file, by default named errs.log.

A number of errors are related to having an incorrect XREF value. An XREF is the internal name used to Identify a family, individual, note, source or other record. An XREF is bracketed by two @ signs. As an example

```
0 @F6@ FAM
1 HUSB @I25@
1 WIFE @I26@
1 CHIL @I17@
```

Here F6 is the internal name of this family. The family refers to other individuals by specifying their XREF values. Also I25, I26 and I17 are XREF values of individuals.

XREF values used within LifeLines are totally under the control of LifeLines. The values that are used are always of the form, a single letter, followed by a number. However, when importing a gedcom LifeLines should accept almost anything as an XREF, converting it to what is needed for internal use. For the curious, the letters that LifeLines uses are I for Individual, F for Family, S for Source, E for Events, and X for other records.

Bad NAME syntax.

A level 1 NAME record must have a value, and the value must not be a pointer, and may have no more than two slashes.

Example 1.13 Example of bad NAME syntax (too many slashes in this case)

```
0 @I99@ INDI
1 NAME Mary /Smith/ nee /Jones/
```

Family F13 has an incorrect key.

Person XYZ has an incorrect key: skipped.

The XREF values must be unique. These error messages are generated if a XREF is found to refer to two different records. For example, the XREF for a family is the same as one for an individual.

Example 1.14 Example of XREF XYZ being defined twice

```
0 @XYZ@ INDI
1 NAME Mary /Smith/ nee /Jones/
0 @XYZ@ FAM
1 HUSB @I48@
```

Person defined here has no name.

Each INDI record must have at least one level 1 NAME record, if the configuration option RequireNames is non-zero.

Person INDI_XYZ is multiply defined: skipped.

No XREF value of an INDI may be reused. This message indicates that INDI_XYZ has been used to identify two different persons.

Example 1.15 Example of person multiply defined

```
0 @INDI_XYZ@ INDI
1 NAME John /Smith/
0 @INDI_XYZ@ INDI
1 NAME Jack /Smith/
```

The family defined here has no key.

Each FAM record must have an XREF value. (The XREF is the letters between "0" and "FAM".)

Example 1.16 Example of family record missing key

```
0 FAM
1 HUSB @I1@
```

Family F123 is referred to but not defined.

A FAMS or FAMC record on an Individual refers to a family but there is no FAM record with that XREF.

The person defined here has no key: skipped.

Each INDI record must have an XREF value. (The XREF is the letters between "0" and "INDI".)

Example 1.17 Example of person record missing key

```
0 INDI
1 NAME John /Smith/
```

This line has a level number that is too large.

A child level must be one higher than its parent level.

Example 1.18 Example of level number which is too high

```
0 @I99@ INDI
1 NAME //
3 NOTE name is missing
```

This FAMS line is missing a value field (INDI I99).

Lineage-linking tags must have a value, including tags FAMC, FAMS, FATH, MOTH, HUSB, WIFE, CHIL.

Example 1.19 Example of lineage-linking line which is missing required value

```
0 @I99@ INDI
1 NAME John /Smith/
2 FAMS
```

1.25 HANDLING SOURCE, EVENT AND USER-DEFINED RECORDS

LifeLines supports source, event and other, user-defined record types. You can access these features in two ways: either through the x operation from the main menu, or via commands in the individual and family browse screens. The first approach might be most convenient when you are solely working with these record types. The second makes it easier to work with source, event and user-defined records in parallel with your person and family records; this can be useful for instance when you want to create references from your person and family record to your source, event and user defined records as you create them, and to view and edit records that you have referenced from within a person or family record.

Using the first of these two possibilities LifeLines displays the following menu:

```
What activity do you want to perform?

  s  Browse source records
  e  Browse event records
  x  Browse other records
  1  Add a source record to the database
  2  Edit source record from the database
  3  Add an event record to the database
  4  Edit event record from the database
  5  Add an other record to the database
  6  Edit other record from the database
  q  Return to main menu
```

Using the second variant (from the person and family browse screens), the following six commands are available. The first three are described alongside with their counterparts in the x menu (they do mostly, but not entirely, the same things); the last three are described separately:

```
  %s  Add source
  %e  Add event
  %o  Add other
```

```
  $s  List sources
  $n  List notes
  $$  List references
```

The handling of source, event and user-defined records in LifeLines is still in development. For example, sources cannot yet be searched by REFN or be deleted.

s Browse source records.

This will bring up a list of all source records by number, showing REFN, title, and author for each. A record may be selected from this list to edit.

e Browse event records.

This will bring up a list of all event records by number. As of yet this does not display any information about each event. (Suggestions as to how to summarize events are welcome). A record may be selected from this list to edit.

x Browse other records.

This will bring up a list of all other records by number, showing the 0 level line as summary. A record may be selected from this list to edit.

1 Add a source record to the database.

This operation (or alternatively **%s** from the person, family or auxiliary browse screens) is used to add a new source record to the database. LifeLines creates a template source and puts you in the screen editor to edit the template. The default template is:

```
  0 SOUR
  1 REFN
  1 TITL Title
  1 AUTH Author
```

Do not change the `0 SOUR` line. Otherwise you may edit this record any way you like. The `1 REFN` line is a special line you can use to give the source a symbolic name that can be used in other records to refer to the source record. See the section on using `REFN` values. Because many sources have a title and an author, the default template has these lines. You may adjust the source template via the SOURREC user option (either at the database level, via the user options, or globally, via the configuration file). Here is how I recorded one of the sources in my database:

Example 1.20 Example of a source record

```
0 SOUR
  1 REFN jcw
  1 TITL The Wetmore Family of America, and its Collateral Branches: with
    2 CONT Genealogical, Biographical, and Historical Notices
  1 AUTH James Carnahan Wetmore
  1 PUBL
    2 DATE 1861
    2 PLAC Albany, New York
    2 INST Munsell and Rowland
    2 ADDR 78 State Street
```

The add source command in the x menu is equivalent to the **%s** command available in the person or family browse screen, in terms of entering the source record; after saving the source record though, there is one significant difference, in that that you then are presented with the following prompt (also, the key value of the new record is displayed in the status bar at this point):

```
Please choose from the following options:
  1: Insert xref automatically at bottom of current record.
  2: Edit current record now to add xref manually.
  3: Browse new record (without adding xref).
  4: Return to current record (without adding xref).

 Commands:   j Move down     k Move up    i Select     q Quit
```

The four options perform the following actions:

1: LifeLines adds a reference to the new source record at the bottom of the person or family record that was visible in the display screen when you prompted LifeLines to create the new source record.

2: LifeLines will open the person or family record that was visible in the display screen when you prompted LifeLines to create the new source record, so that you can manually enter a reference to the newly created source record (perhaps as a source reference to an event that you are planning to add).

3: LifeLines lets you browse (and optionally reopen for editing) your newly added source record.

4: You are returned to the person or family record that was visible in the display screen when you prompted LifeLines to create the new source record. No reference is added to the new record.

2 Edit source record from the database.

Use this operation to edit an existing source record already in the database. When you select this operation LifeLines asks you to identify a source:

```
Which source record do you want to edit?
enter key or refn:
```

Identify a source by entering its key value, with or without the leading S, or by entering its `REFN` value. LifeLines retrieves the record and puts you in the editor with the record.

3 Add an event record to the database.

This operation (or alternatively **%e** from the person, family or auxiliary browse screens) adds a new event record to the database. LifeLines creates a template event and puts you in the screen editor to edit the template. The default template is:

```
0 EVEN
  1 REFN
  1 DATE
  1 PLAC
  1 INDI
    2 NAME
    2 ROLE
  1 SOUR
```

You may adjust the event template via the EVENREC user option (either at the database level, via the user options, or globally, via the configuration file).

Do not change the `0 EVEN` line. Otherwise you may edit this record any way you like. The `1 REFN` line allows you to give this event a symbolic name you can use when you want to refer to this event from other records. See the section on using `REFN` values. The default template suggests that an event has a date, a place, and refers to persons in roles with respect to the event. There is far less experience with event-based GEDCOM than there is with simple person and family GEDCOM. You may even be wondering why you would need event records when you can simply tuck events away in person and family records. This is a topic that may get covered in an appendix.

The add event command in the x menu is equivalent to the **%e** command available in the person or family browse screen, in terms of entering the event record; after saving the event record though, there is one significant difference, in that that

you then are presented with the following prompt (also, the key value of the new record is displayed in the status bar at this point):

```
Please choose from the following options:
  1: Insert xref automatically at bottom of current record.
  2: Edit current record now to add xref manually.
  3: Browse new record (without adding xref).
  4: Return to current record (without adding xref).

 Commands:   j Move down     k Move up    i Select     q Quit
```

The four options perform the following actions:

1: LifeLines adds a reference to the new event record at the bottom of the person or family record that was visible in the display screen when you prompted LifeLines to create the new event record.

2: LifeLines will open the person or family record that was visible in the display screen when you prompted LifeLines to create the new event record, so that you can manually enter a reference to the newly created event record.

3: LifeLines lets you browse (and optionally reopen for editing) your newly added event record.

4: You are returned to the person or family record that was visible in the display screen when you prompted LifeLines to create the new event record. No reference is added to the new record.

4 Edit event record from the database.

Use this operation to edit an existing event record from the database . When you select this operation LifeLines asks you to identify an event:

```
Which event record do you want to edit?
enter key or refn:
```

You identify a event by entering its key value, with or without the leading E, or by entering its `REFN` value. LifeLines retrieves the record and places you in the screen editor with the record.

5 Add an other record to the database.

This operation (or alternatively **%o** from the person, family or auxiliary browse screens) adds a new user-defined record to the database. LifeLines creates a template and puts you in the screen editor to edit the template. The default template is

```
0 XXXX
1 REFN
```

Replace *XXXX* with the tag string you select for the type of the new record. You are free to choose any tag value except `INDI`, `FAM`, `SOUR` and `EVEN`. For example, if you keep record information about the ships that your North American immigrant ancestors arrived on, you would keep records about those ships in your database; the tag `SHIP` suggests itself for such records. The `1 REFN` line allows you to give this record a symbolic name you can use when you want to refer to it from other records. See the section on using `REFN` values. You may adjust the other template via the OTHR user option (either at the database level, via the user options, or globally, via the configuration file).

The add other record command in the x menu is equivalent to the **%o** command available in the person or family browse screen, in terms of entering the other record; after saving the record though, there is one significant difference, in that that you then are presented with the following prompt (also, the key value of the new record is displayed in the status bar at this point):

```
Please choose from the following options:
  1: Insert xref automatically at bottom of current record.
  2: Edit current record now to add xref manually.
  3: Browse new record (without adding xref).
  4: Return to current record (without adding xref).

 Commands:   j Move down     k Move up    i Select     q Quit
```

The four options perform the following actions:

1: LifeLines adds a reference to the new other record at the bottom of the person or family record that was visible in the display screen when you prompted LifeLines to create the new other record.

2: LifeLines will open the person or family record that was visible in the display screen when you prompted LifeLines to create the new other record, so that you can manually enter a reference to the newly created other record (perhaps as a note reference to an event that you are planning to add).

3: LifeLines lets you browse (and optionally reopen for editing) your newly added other record.

4: You are returned to the person or family record that was visible in the display screen when you prompted LifeLines to create the new other record. No reference is added to the new record.

6 Edit other record from the database.

Use this operation to edit an existing user-defined record from the database. When you select this operation LifeLines asks you to identify the record:

```
What record do you want to edit?
enter key or refn:
```

You identify a record by entering its key value, with or without the leading X, or by entering its `REFN` value. LifeLines retrieves the record and places you in the screen editor with the record.

$s List sources

This command is available in the person and family browse screens. It will bring up a list of all source records that are referenced within the currently displayed person or family record, by the order they appear in the person/family record, showing REFN, title, and author for each. A record may be selected from this list to view in the auxiliary browse screen (and optionally be edited from there).

$n List notes

This command is available in the person and family browse screens. It will bring up a list of all note records that are referenced within the currently displayed person or family record, by the order they appear in the person/family record, showing REFN, title, and author for each. A record may be selected from this list to view in the auxiliary browse screen (and optionally be edited from there).

$$ List all references

This command is available in the person and family browse screens. It will bring up a list of all records that are referenced within the currently displayed person or family record, by the order they appear in the person/family record, showing REFN, title, and author for each. A record may be selected from this list to view in the auxiliary browse screen (and optionally be edited from there).

1.26 FAMILY STRUCTURE AND MERGING PERSONS AND FAMILIES

LifeLines 3.0.2 has relaxed most of restrictions on family structure that were imposed by earlier versions. For example, a family record may have more than one parent/spouse of the same sex; a person may be a child in more than family. This is a controversial issue. Some users insist that family relationships should imply biological relatedness, and that all other relationships should be handled by different means. Others insist that non-traditional families (any number of parents/spouses of any sex) should be allowed, and that children can be members of more than one family (eg, natural family and adoptive family). LifeLines no longer takes a position on this matter; you are free to set up families any way you like; the operations that add spouses and children to families no longer check for non-traditional arrangements. It is possible that a future release will include a user option to either disallow or to ask for confirmation about non-traditional relationships.

LifeLines provides features for merging persons together and for merging families together. The person merging feature is accessed from the tandem person browse mode, and the family merging feature is accessed from the tandem family browse mode. You browse to the two persons or families you want to merge and then use the j command. Merging is necessary when you discover that two or more person records, or two or more family records, represent the same person or family, respectively.

Versions of LifeLines prior to 3.0.2 required that persons and families meet certain criteria before they could be merged. The criteria ensured that the merged persons and families would still meet traditional family structuring rules. With the relaxation of the structuring rules, restrictions on merging have also been removed. It is now possible to create non-traditional relationships by

merging traditional persons and/or families. For example, if you merge two persons that happen to be children in two different families, the merged person will be a child in both families. If you want to maintain only traditional relationships in your database you may have to makes further to changes to relationships after you complete a merge operation.

1.27 LINKING RECORDS TOGETHER AND USING THE REFN FEATURE

Records in a LifeLines database may refer to other records via cross-reference links. The lineage-linked references are maintained directly by LifeLines through operations found in the browsing mode menus. These references are the links from a person to families (`1 FAMC` and `1 FAMS`), and the links from a family to persons (`1 HUSB`, `1 WIFE` and `1 CHIL`). Because LifeLines maintains these links you are not allowed to change these lines when you are editing records. There are a couple of seeming exceptions to this rule. For example, you may change the order of `1 CHIL` lines in a family record in order to change the order of children in a family, and you may change the order of `1 FAMS` lines in a person record to change the order of families the person was a spouse or parent in. These operations are allowed because they don't affect which person records refer to which family records and vice versa.

Besides the lineage-links that are maintained by LifeLines, you may place your own links in records. Probably the most common example of this is referring events within a person record to the record of the information source for the event. For example:

Example 1.21 Example of referring events from a person record

```
0 @I23@ INDI
  1 NAME Thomas/Whitmore/
  1 BIRT
    2 DATE about 1615
    2 PLAC England
    2 SOUR @S3@
...
0 @S3@ SOUR
  1 REFN cat
  1 TITL New England Marriages Prior to 1700
  1 AUTH Clarence Almon Torrey
...
```

The `2 SOUR @S3@` line in the person record refers to the source record. LifeLines allows any specific structure within a record (in this case a birth event) to refer to another record. It is not possible to refer to a specific location within another record, though this may be supported eventually.

This example implies that when linking one record to another you must know the key of the target record (S3 in the example). This is not desirable because internal record keys may change when the records are exported from one database or imported to another.

Because internal key values are not permanent, LifeLines allows you to assign a permanent user-defined key to any record in the database using the `1 REFN` line. The value of this line is a string that you choose as your permanent key value for the record. When adding a link to a record that has a user `REFN` key value, you may use that value instead of the internal key value. For example, when adding the person in the previous example you could edit the new record as follows:

Example 1.22 Example of adding a new person

```
0 INDI
  1 NAME Thomas/Whitmore/
  1 BIRT
    2 DATE about 1615
    2 PLAC England
    2 SOUR <cat>
```

Instead of using the actual key value of the source, S3, the `REFN` value cat was used. The `REFN` value must be enclosed by angle brackets when used this way. LifeLines automatically replaces the `REFN` link with the proper internal key value when the record is stored in the database.

The REFN value may also be used when searching for person, source, event and user-defined records. You should not add more than one REFN line to a record, and every REFN value should be unique.

1.28 LIFELINES REPORTS

Lifelines comes with some prewritten reports, in the reports directory. See the file index.html in the reports directory, for a list and summary of these reports. Lifelines also has a built-in report language for writing your own reports, and comes with a report manual describing how to write reports: see the separate lifelines report manual (ll-reportmanual.html, or ll-reportmanual.pdf).

1.29 System and User Properties

There are a number of properties that can be specified to customize the behavior of LifeLines. These properties can be specified in LifeLines configuration files, in each LifeLines database or in some cases by environment variables.

System Properties are properties that have a predefined meaning to lifelines, such as LLEDITOR (see its meaning below). User Properties typically have no predefined meanings as they are simply a string that a report looks up in the property tables. It can be anything a user desires. To simplify report writing a number of User Properties are predefined with specific meanings. These User Property Names begin with 'user.' and are listed below. For example, many reports have abstracted the concept of the user's name to the property user.fullname. By defining this property in your llines startup file, it allows a report to reference your name as the source of the data being printed without having it hard-coded in the report.

When LifeLines begins execution, it reads any specified configuration files and extracts Properties from the files read. It is possible for multiple configuration files to be read. Properties defined in these files will be stored in the global property table. If multiple definitions of the same property are seen, the latest definition overrides prior definitions. Configuration files are read as follows:

Note:	If a configuration file defines LLCONFIGFILE, it is not entered in the global table, but the value of this parameter is read as a configuration file after completion of reading the current file.
Note:	In the following, the name of a user configuration file is listed as .linesrc. On windows versions of lifelines this name is lines.cfg.
1	If a configuration file name is passed to the program with the -C option, or if LLCONFIGFILE is defined as an environment variable use the value supplied as the name of the configuration file. Do not read configuration information from the files listed in 2a, 2b, 2c, or 2d.
2a	If the file /etc/lifelines.conf exists, read it. '/etc' is a placeholder for the standard location for system configuration files as defined when lifelines was built. It is often /etc or /usr/local/etc.
2b.	If the file $HOME/.linesrc exists, load parameters from it.
2c.	If the file .linesrc exists in the current directory read parameters from it.
2d	If LLCONFIGFILE is defined in the command line or database table and its value is the name of a file, load parameters from that file.

When LifeLines searches for a property it looks for it as follows:

cmdline table

First, LifeLines looks in the cmdline table. This table contains the values that have been specified using the -I option to LifeLines.

database table

Next, LifeLines looks in the database table. This table contains the values which have been stored in the current database.

global table

Next, LifeLines looks in the global table. This table contains the values found when reading in the configuration files.

environment table

Finally, LifeLines looks in the environment table. Only a handful of properties may be set from the environment, namely LLPROGRAMS, LLREPORTS, LLARCHIVES, LLDATABASES, and LLEDITOR.

Properties are named group.subgroup.property or group.property, or even just property. The following keys are available at the moment:

LLPROGRAMS

`LLPROGRAMS` is the search path for LifeLines report generating and other programs.

LLREPORTS

`LLREPORTS` is used to select a directory where all generated reports and program outputs will be placed.

LLARCHIVES

`LLARCHIVES` is used to select a directory where all database backup files will be stored

LLDATABASES

`LLDATABASES` is a list of directories that contain database directories used to locate database directories themselves

LLNEWDBDIR

`LLNEWDBDIR` a single directory where new databases specified without paths will be created. This is a convenience for users who generally put all their databases under a single common directory.

LLEDITOR

Overrides the default screen editor

Codeset Information:

For the following parameters related to codeset, the values are a String denoting code set in use in data. Special handling is provided for UTF-8, which may be entered as "UTF-8", "utf-8", or "65001". (The official, and preferred, name is UTF-8.

codeset

Is a property that report programs can read whose value is the codeset of the current database.

GedcomCodeset

Specifies the default Codeset to use when processing Gedcom files. There are actually 3 properties. *GedcomCodesetOut* and *GedcomCodesetIn* can be used to specify the codeset for output or input. If either of these is not specified *Gedcom-Codeset* is used. Most Gedcom files should contain a codeset property, so this is not usually relevant when reading a Gedcom file, unless the input Gedcom file lacks a codeset property. This is used, however, when writing out a Gedcom file. NB: This is ignored if the database has no internal codeset specified.

EditorCodeset

Specifies the codeset to use for editing. There are actually 3 properties. *EditorCodesetOut* and *EditorCodesetIn* can be used to specify the codeset for output or input. If either of these is not specified *EditorCodeset* is used. This property allows lifelines to convert from its internal codeset to the one you use in your editor, so this is important when your editor does not use the same codeset as your database. NB: This is ignored if the database has no internal codeset specified.

ReportCodeset

Specifies the codeset to use for reports. There are actually 3 properties. *ReportCodesetOut* and *ReportCodesetIn* can be used to specify the codeset for output or input. If either of these is not specified *ReportCodeset* is used. Recently created reports may actually specify their codeset, in which case this is not used, but for reports which do not include a specification of codeset, the *ReportCodesetIn* (or *ReportCodeset*) specifies how the report will be understood. In any case, the output of a report program will be written in the codeset given by *ReportCodesetOut* (or *ReportCodeset*). NB: This is ignored if the database has no internal codeset specified.

GuiCodeset

Specifies the codeset to use for interaction with the graphical user interface. There are actually 3 properties. *GuiCodesetOut* and *GuiCodesetIn* can be used to specify the codeset for output or input. If either of these is not specified *GuiCodeset* is used.

LongDisplayDate

Six comma separated numbers. Specifying the format for days, months, years, date format, era format, and complex format. If string does not contain 6 comma separated numbers all formats are set to 0, except date format is set to 14. These formats are used by LifeLines to display dates in long format.

LongDisplayDatePic

a custom ymd date picture. A string in which %y, %m and %d will be converted to the corresponding year, month and day. This picture is used by LifeLines to display dates in long format.

ShortDisplayDate

Six comma separated numbers. Specifying the format for days, months, years, date format, era format, and complex format. If string does not contain 6 comma separated numbers all formats are set to 0, except date format is set to 14. These formats are used by LifeLines to display dates in short format.

ShortDisplayDatePic

a custom ymd date picture. A string in which %y, %m and %d will be converted to the corresponding year, month and day. This picture is used by LifeLines to display dates in short format.

LocaleDir

run-time specification of locale directory

NewDbProps

if specified, properties to add to a newly created database. The string is of the form "option1=value1\noption2=value2".

IllegalChar

Apparently not used at this time.

DenySystemCalls

If set to a non-zero value will disable use of the `system` function.

UiLocaleCollate

ui collating sequence

UiLocaleMessages

ui messages locale

RptLocaleCollate

Report Collating Sequence

RptLocaleMessages

Report Messages Locale

ShortOmitString

If defined this string replaces characters at the end of an event being printed.

DisplayKeyTags

If set to 1 an i or f is prepended to individual and family keys when shown on-screen.

AnnotatePointers

If set to 1 a descriptive comment is added to pointers when editing family or individual records to help identify who is being referred to. These tags are removed when the edit is finished and not stored in the database.

TTPATH

list of directories to search for translation table files (*.tt)

TTPATH.debug

if defined, LifeLines outputs debug information related to translation table processing.

InputPath

path to look for in when reading in gedcom files

INDIREC

The default template to use whenever creating a new individual. If not specified, the concatenation of the following is used:

```
"0 INDI\n1 NAME Fname/Surname\n1 SEX MF\n"
"1 BIRT\n  2 DATE\n  2 PLAC\n"
"1 DEAT\n  2 DATE\n  2 PLAC\n1 SOUR\n"
```

FAMRECBODY

The default template to use whenever creating a new family. If not specified, the following is used:

```
"1 MARR\n  2 DATE\n  2 PLAC\n  2 SOUR\n"
```

Note: Unlike the others, this should not include the 0 level FAM tag.)

SOURREC

The default template to use whenever creating a new source. If not specified, the following is used in English:

```
"0 SOUR\n1 REFN\n1 TITL Title\n1 AUTH Author"
```

EVENREC

The default template to use whenever creating a new event. If not specified, the following is used in English:

```
"0 EVEN\n1 REFN\n1 DATE\n1 PLAC\n1 INDI\n  2 NAME\n  2 ROLE\n1 SOUR"
```

OTHR

The default template to use whenever creating a new other record. If not specified, the following is used in English:

```
"0 XXXX\n1 REFN"
```

HDR_SUBM

string to use when writing gedcom files for the SUBM. Default is

```
"1 SUBM".
```

HDR_GEDC

String containing the GEDC block for the header record when exporting GEDCOM. The default is

```
"1 GEDC\n2 VERS 5.5\n2 FORM LINEAGE-LINKED".
```

HDR_CHAR

Normally lifelines will populate the CHAR block in the header record automatically when exporting GEDCOM, with the actual character encoding being used in the export; this can be found in the settings displayed in the u(ser) c(harset settings) page. However, if desired, the HDR_CHAR variable may be used to overwrite the entire CHAR line. String containing the CHAR block for the header record when exporting GEDCOM. An example HDR_CHAR value would be "1 CHAR ASCII".

ReportLog

name of file to print report program errors to, if not specified use stdout curses window.

CrashLog_llexec

file to write crash log into for llexec. If not set, log is written to the file `CrashLog_llexec.log`.

CrashLog_llines

file to write crash log into for llines. If not set, log is written to the file `CrashLog_llines.log`.

LLTTEXPORT

directory to write translation tables to. Default is the current directory.

ImportLog

where to log errors found while importing a gedcom file. default is errs.log.

gettext.path

for windows, if specified path to (re)load gettext dll

iconv.path

for Windows, path to link dynamically to gettext and iconv

user.fullname

The full name of the current user. If not found as a property it is fetched from the system when possible.

user.email

The email address of the current user. If not found as a property it is fetched from the system when possible.

user.address

Postal address of the current user.

user.phone

The phone number of the current user.

user.url

URL to the users home page.

1.30 Other Executables

1.30.1 dbverify

dbverify is a utility to check an existing database and report on various inconsistencies. It can also repair a number of issues found.

dbverify supports the following options:

-a	Perform all checks (does not include fixes)
-D	Fix bad delete entries
-e	Check events
-f	Check families
-F	Alter any bad family lineage pointers (to _badptr)
-g	Check for ghosts (names/refns)
-G	Check for and fix ghosts (names/refns)
-h	Display help text (this text)
-i	Check individuals
-l	Check database structure
-m	Check for records missing data entries
-M	Fix records missing data entries
-n	Noisy (echo every record processed)
-s	Check sours
-x	Check others

1.30.2 btedit

This program edits raw binary btree blocks in a LifeLines lifelines database file. Some information about its usage is provided by running:

```
btedit -h
```

Do NOT use this unless you know what you are doing and you have backed up your database.

The LifeLines database has proven rather robust over the years. However, it is important to make frequent backups of any database. If you experience database corruption, make sure you save a copy of the database before trying any recovery process. Only work on a copy of the data so that the data is not further damaged.

Try exporting the database to a gedcom file from within **llines**. Compare the saved file with previously saved versions. Editing the gedcom file to correct issues is often easier than using btedit.

1.30.3 llexec

llexec invokes the LifeLines report execution program without a GUI for non-interactive processing of report programs. The supported options are the same as for **llines**.

An example usage, which will open the database 'myfamily' and run the report eol.ll is:

```
llexec myfamily -x eol
```

Programs that require input, will prompt for that data and read from standard input. If a program required the input of a 0 or 1 to control the output, the following is one way to allow use of llexec in a script:

```
echo "1" | llexec myfamily -x myprog
```