GENERALIZED ITERATIVE RECORD LINKAGE SYSTEM

Ted Hill and Francis Pring-Mill, Statistics Canada

ABSTRACT

The Generalized Iterative Record Linkage System (GIRLS) project was initiated at Statistics Canada in 1978. This paper outlines the concepts behind the system, and summarizes how these have been implemented to provide a powerful tool suitable for a variety of record linkage applications.

1.0 RECORD LINKAGE AND GIRLS

Record linkage is the process of identifying two or more records which refer to the same entity. An entity could be a person, or a business, for example.

In the case where records have unique identifiers (for example, social insurance number), the process of linking is relatively simple as it involves matching on only one field. However in cases where records do not have unique identifiers, information from several fields typically has to be compared to estimate the likelihood that a potential link is a 'true' one. For these cases record linkage is a probabilistic process, and it is for this situation that GIRLS was designed.

GIRLS stands for the "Generalized Iterative Record Linkage System" which has been developed at Statistics Canada, starting in 1978. Since then, the system has been systematically maintained and enhanced on a regular basis.

GIRLS provides a command language in which you can write your own rules for comparing records. Statistically-derived weights are attached to potential links according to the outcomes of these comparisons. Your GIRLS commands are automatically translated into PL/1 (a high-level programming language), compiled, link-edited and executed on the input files to generate an online project database of potential links and the records involved in them. Using other GIRLS commands, you can then query this database to see the results. If these are not satisfactory, you can update the database in various ways, or simply change your comparison rules and try again.

To this end, GIRLS breaks the linkage process into a sequence of distinct phases. Each phase involves deciding on values for system parameters, examining their effect, and adjusting the values as necessary before going on to the next phase. Results from later phases often suggest adjustments to earlier phases. Because phases are distinct, you can easily retrace your steps, run an earlier phase again with new adjustments, run intermediate phases as they are, and quickly catch up to where you were. This is why GIRLS is called an 'iterative' record linkage system.

The principal aims of GIRLS are:

1. To enable you to develop the best comparison rules and statistical weights for the purpose of your linkage project.
2. To provide a convenient framework for this development.
3. To encourage iterative refinement through a sequence of phases and reports.
4. To make the final linkage fast, cheap, and accurate.

Examples of GIRLS applications include:

1. 'Follow-up' studies

   Health Division at Statistics Canada currently runs linkage projects with files provided by employers of individuals exposed to potential health hazards in the course of their work (e.g. uranium miners). These are linked with the Canadian Mortality Database to check that the proportion of matches found is not above normal.

   Such studies can detect risks to health associated with particular occupations, thus pointing the way to causes of disease. They can also aid in testing the long-term effects of curative measures.

2. Building 'case histories'

   Separate records referring to the same person often accumulate in a system. For example, a new record is often made each time an individual is admitted to a hospital. GIRLS can conveniently bring these records together, enabling larger composite records to be made representing 'case histories' for individuals.

2.0 FEATURES OF GIRLS

In the past, record linkage systems have usually been tied to methodologies suited to particular application requirements. GIRLS provides a general solution to developing particular linkage systems.

Its principal features are:

1. A sequence of phases encourages iterative refinement of the linkage process.
2. The full power of database management technology is provided. This includes: automatic maintenance of data integrity across separate files, checkpointing facilities for project recovery, as well as back-up and restore procedures.
3. Both 'one-' and 'two-' file linkages can be performed. (One-file, or internal, linkages can be useful for unduplicating a file or creating composite records.)
4. A variety of samples of records from the input files can be extracted for the purposes of experimenting.
5. A simple but powerful GIRLS command language is provided to write comparison rules, update the project database, and obtain a wide variety of reports at many levels of detail.
6. The commands provided for writing comparison rules can detect full agreement, various levels of partial agreement, disagreement, and missing values. They can also specify cross comparisons of different fields, as well as rules to be executed conditional on the outcomes of previous comparisons.
7. For special purposes you can also write your own PL/1 code and have it included in the Compare program automatically generated from your GIRLS commands.

8. Statistically-derived weights are generated and attached to links to reflect the probability that the records being compared refer to the same entity.

9. Potential links are automatically classified as: rejected, possible, or definite, by comparing link weights against threshold values. You specify these threshold values, and you can easily adjust them. You can also reclassify links manually.

10. Records which refer to the same entity are grouped. Where conflicts exist within groups, these can be resolved either automatically by the system, or manually on a record-by-record basis. (For example, a conflict would exist when records are expected to link to at most one record on the 'other' file, but a group contains some which have linked to several records.)

11. Both batch and online modes are available. Online enables fast iterative adjustment of system parameters by providing quick feedback as to the current state of the project database.

3.0 BASIC OPERATIONS

The phases of the GIRLS system can be grouped into three main operations.

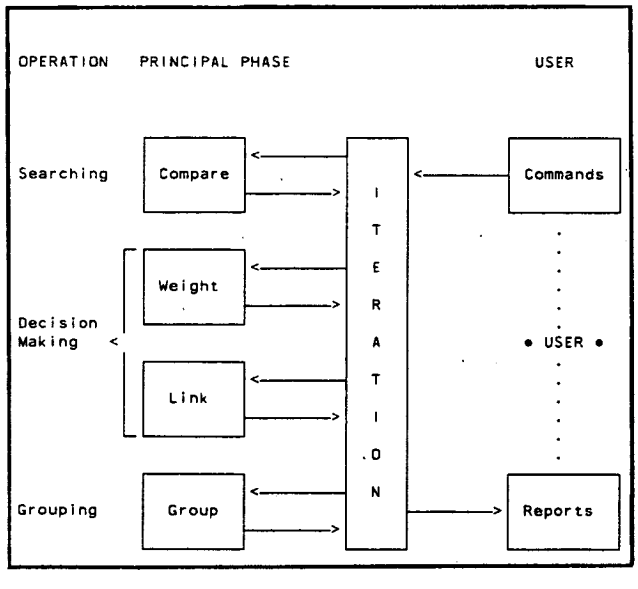
1. Searching.

2. Decision Making.

3. Grouping.

This is shown below:

Figure 1: Basic operations



3.1 Searching

In this operation, pairs of records are compared field by field according to comparison rules you specify. Theoretically, every possible pair of records should be compared. However the number of possible pairs in even a small file is very large. So for practical reasons, records are first blocked into smaller 'pockets' in such a way that it is realistic to look for links only within pockets.

You use GIRLS commands to define your input files, indicate which fields define your pockets, select your sample of records, and specify rules according to which your records are to be compared. Your GIRLS commands are then automatically translated into a PL/1 program, called the Compare program, which is executed on your input files to produce the project database of potential links.

You can write rules to compare fields with values that are: character (e.g. surname), numeric (e.g. birthyear), or coded (e.g. for fields with a small number of discrete values such as birthplace). Your rules can be made conditional on particular outcomes from previous comparisons. You can also specify cross comparisons of different fields (for example, first given name with second given name, in the event that straight comparisons of each field have not already produced an agreement). If your rules do not fit conveniently into the format of the GIRLS command language, you can also write them yourself in PL/1 and have them included in the Compare program.

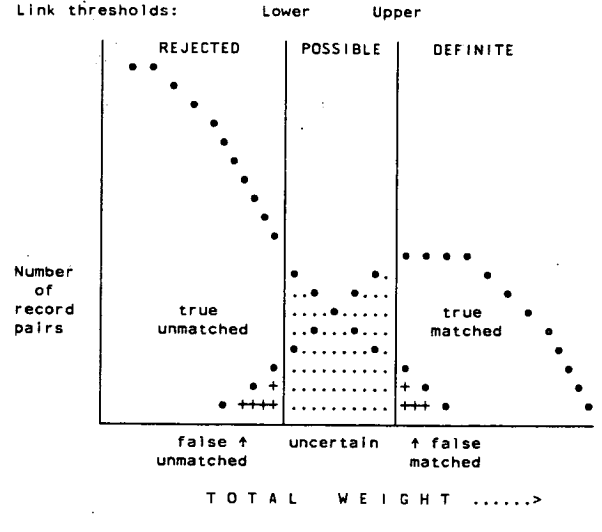
The outcome of having executed a comparison rule can be: agreement, one of various levels of partial agreement, disagreement, or missing. You can specify a 'global' weight to be attached in the event of each one of these possible outcomes.

3.2 Decision Making

In this operation, the potential links generated by the Compare program are evaluated. This involves updating link weights and comparing them against threshold values to decide which to keep and which to reject. Link weights are updated with 'frequency weight sets' which reflect the probability of particular agreements happening by chance. These weights are derived according to formulae developed by Geoff Howe[1], Mike Eagen, and David Binder from methodologies proposed by Howard Newcombe[2], Ivan Fellegi and Alan Sunter[3].

After weight update, the status of links is determined by comparing their total weights against two threshold values. Links with weights above the upper are classified as 'definite', those with weights below the lower threshold are 'rejected', those with weights between the two are 'possible'. This is shown in Figure 2, which is explained as follows:

Figure 2: Link thresholds classify links into three statuses



Let all possible record pairs be divided into two populations: those record pairs which are 'truly matched', and those which are 'truly unmatched'. The goal of the linkage project is then to find the members of the 'truly matched' population. Because it represents all possible record pairs which do not match, the true unmatched population will be far greater than the true matched one. This is shown on the left. The smaller true matched population is shown on the right. The problem is the overlap in the middle, because for these record pairs it is not obvious to which distribution they belong.

The two threshold lines show how GIRLS handles this problem area. Links to the right of the upper threshold are considered 'definite', those to the left of the lower are considered 'rejected', those between the two are considered 'possible'. While permitting flexibility, this approach allows two types of error which any linkage project should aim to minimize.

First is the 'false unmatched' area on the left. These are the record pairs which have been rejected even though they were part of the true matched population. This can happen when information is incomplete or inaccurate on records which 'should' have matched. Second is the 'false matched' area on the right. These are the record pairs which have been accepted even though they were part of the true unmatched population. This can happen when records look very similar even though they refer to different entities, e.g. the different members of the same family. At first glance, these two areas can be minimized simply by setting the thresholds far apart. However this makes for many possible links in between, which will have to be resolved later. By adjusting the thresholds and inspecting various samples of links, however, you can choose the best thresholds for your purposes.

3.3 Grouping

In this operation, the records are grouped according to the status of the links between them. Records may have just one link to another record, or they may have several links to several records. Records joined either by possible or definite links are arranged into 'major' groups - which can be large. Within major groups, records joined by definite links are further arranged into 'minor' groups. A major group may therefore contain several minor groups, and it is the minor groups that contain the best links.

At this stage, 'conflicts' may arise, typically when groups are larger than you want them to be. The system identifies conflicts for you based on your linkage requirement, e.g. one-to-one (i.e. groups are to contain pairs of records only, one from each file). Resolving the conflicts can be done in either of two ways, or both:

1. You can let the system resolve conflicts automatically. This is called 'automatic resolution'. In this case all you specify is your linkage requirement, e.g. one-to-one, many-to-one, or one-to-many.

2. You can resolve the conflicts yourself manually. This is called 'manual resolution'.

You can also use both methods, automatic resolution first followed by an examination of the results and some manual rearrangement where necessary.

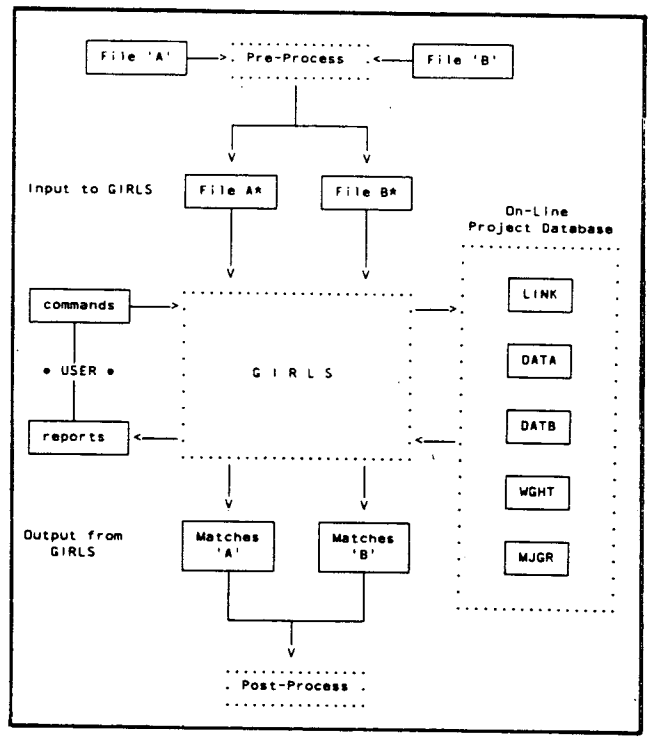
4.0 ENVIRONMENT

4.1 Flowchart

Figure 3 shows a flowchart overview of the system. At the top, two files of records (File 'A' and File 'B') are pre-processed for input to GIRLS. In the middle, records are compared according to your comparison rules, and an online project database is created on the right. This consists of potential links (LINK), the records involved in them (DATA and DATB), together with other files for use later.

On the left, the user is shown interacting with the system via GIRLS commands in the light of the linkage project requirements and feedback from reports as to the current state of the project database. At the bottom, two files of 'matches' are produced. On each output file, each original input record that has been linked is identified by a unique sequence number and has a number identifying the group to which it has been assigned.

Figure 3: Flowchart overview of the system



4.2 Iteration

Iterative refinement of the linkage process can include adjustments to:

1. *COMPARISON RULES*

   From the very many possible links which exist between all possible record pairs, these rules determine which are to be considered the 'potential' links to be written to the project database. These rules can be written, re-written, ordered and re-ordered, so as to produce enough suitable links as efficiently as possible.

2. *WEIGHTS*

   These are attached to links via the comparison rules which applied to the records when the links were formed. It is easy to modify these weights, and thereby select the best ones for your purposes.

3. *THRESHOLD VALUES*

These determine the proportion of definite, possible, and rejected links. The best mixture depends on the aim of a particular linkage project, and is determined by experimenting with the thresholds, and seeing the types of groups which are formed.

For example, for a statistical study it may be satisfactory to find 90% of the links. While for other types of study, it may be necessary not to miss any of them.

4.3 GIRLS Project Files

Making the iterative concept work in practice requires maintaining data integrity across several files when any one of them is being updated. For this reason, an integrated database approach has been taken using the RAPID Database Management System developed at Statistics Canada.[4] The principal RAPID files are:

1. *WEIGHT FILE (WGHT)*

   For each field to be weighted, this contains the values for the field and the frequency weight for each value.

2. *LINK FILE (LINK)*

   For each 'potential' link between a pair of records, this file contains: - the outcomes (agree, disagree) for each comparison rule - the current total weight of the link - the current status of the link (definite, possible, or rejected) - other system control information

3. *DATA FILES (DATA, DATB)*

   These contain the records involved in potential links.

4. *MAJOR GROUP FILE (MJGR)*

   This contains information for each group, enabling reports to be made according to type of group, e.g. "display all groups having more than six records".

4.4 Typical Scenario

A typical (abbreviated) scenario for a GIRLS linkage project might be:

1. Write rules specifying how fields are to be compared.
2. Calculate frequency weight sets (a SAS function is provided to do this job).
3. Use sampling facilities to select a sample of records from the pre-processed input files.
4. Adjust appropriate system parameters, both in batch mode and/or online, until satisfactory results are obtained.
5. Run the full linkage in batch.

Using the system online greatly speeds up the iterative adjustment of linkage parameters. The result can be a linkage process uniquely adapted to the purposes of your linkage project.

Favourable reports from current users include:

- The system is 'comfortable' to use because you remain in control at all stages.
- The command language enables both updates to be made easily, and reports to be obtained to verify intended results.
- Iteration can be continued for as long as it takes for you to be satisfied.

5.0 PHASES

This section briefly outlines the various phases of the GIRLS system. Further details are given in the Strategy Guide and in the User Guide.

5.1 Pre-Process

Purpose: to get files ready for linking

- standardize names and addresses
- validity check
- decide on POCKET
- assign SEQUENCE numbers. (These uniquely identify each record.)
- make duplicate records, when you know records match although they look different. E.g. a record for an individual using her maiden name, and another record for the same individual using her married name.
- recode, e.g. from different codes to common code. (For example, from one hospital coding system to another.)
- encode, e.g. from surname to NYSIIS code
- split files, e.g. by sex, year
- sort files by POCKET

5.2 Weight Creation

Purpose: to create global and frequency weights

- use the provided SAS function to:
  - calculate frequency weights themselves
  - generate GIRLS weight update commands
  - calculate global weights (optional)

"The rarer the value, the higher the FREQ weight."

The frequency weight formula used is:

$$FW_i = 10 \times \log_2 \left(\frac{\text{total number of records}}{\text{No. occurrences of field value } i} \right)$$

where "FWi" is the frequency weight for value "i". For example, the value "SMITH" for the SURNAME field could have a frequency weight of "40".

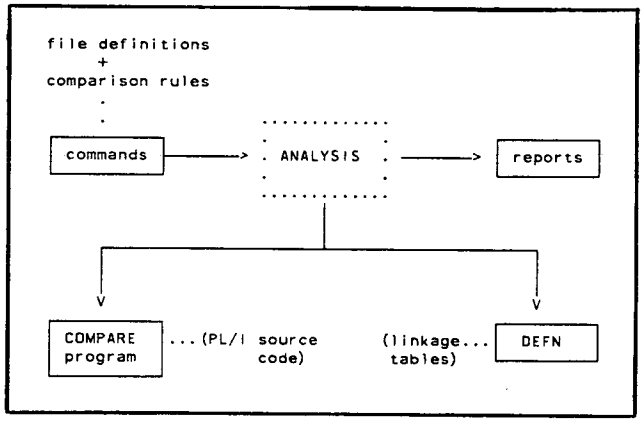
5.3 Analysis

Purpose: to specify comparison rules

- define input files
- choose fields to compare
  - character e.g. surname
  - numeric e.g. birthyear
  - coded e.g. marital status
  - conditional and cross comparisons
  - your own PL/1 code

  choose possible outcomes to weight
  - fully, partially agree
  - disagree
  - missing
- your rules are then translated into a PL/1 program called the 'Compare' program

Figure 4: The Analysis phase



5.4 Compare

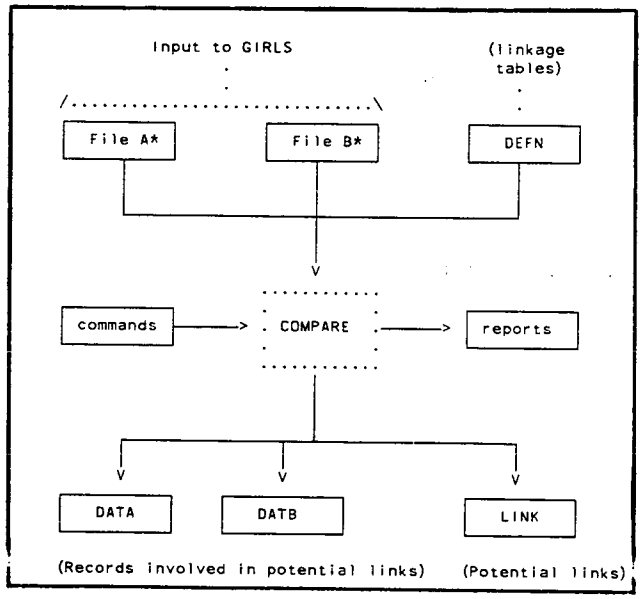
Purpose: to build the linkage database

- set thresholds: upper, lower, and cutoff so as to reject obvious non-matches quickly
- select a sample of pockets with which to experiment
- execute the Compare program

  The comparison rules start assigning global weights to potential links, which are rejected as soon as either current total weight falls below cutoff or if final total weight will be less than the lower threshold.

  The linkage database of potential links and all records involved in them is created.

Figure 5: The Compare phase



5.5 Weight Update

Purpose: to apply and/or modify the weights

- look at link weights 'before'
- apply weights

  You attach frequency weight sets to comparison rules. The system finds all links to which each rule applies and updates the link weights accordingly.
- look at link weights 'after'

5.6 Link

Purpose: to assign statuses to the links

- set a lower and an upper threshold

  The system classifies links by comparing their total weights against these thresholds and assigning a status of definite, possible, or rejected (as explained in Section 3.2).
- inspect results

5.7 Group

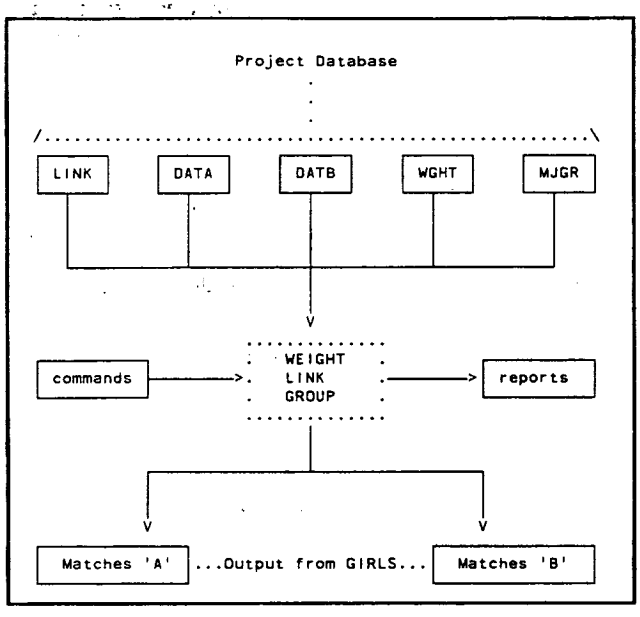
Purpose: to build groups of records

- the system builds 'major' and 'minor' groups of records based on their link status.
  - major groups have both definite and possible links
  - minor groups have definite links only
  - i.e. minor groups contain the best links.
- the system combines groups which share duplicated records. For example, combining a group which contains Mary Smith (maiden) with a group which contains Mary Brown (married).

- resolve group conflicts, either automatically or manually
- output final versions of groups

The Weight, Link, and Group phases are represented below.

Figure 6: The Weight, Link, and Group phases



5.8 Post-Process

Purpose: to use the results of GIRLS

- e.g. for an internal linkage, prepare composite records to represent case histories
- e.g. for a two-file linkage, for each group, generate one record to represent all the members
- create summary files

6.0 EXAMPLE

This is a simple example to show how the GIRLS linkage process works for a two-file linkage.

Part 1 of Figure 7 represents the contents of two files to be linked by GIRLS. File DATA contains 6 records which are to be matched against the 9 records of file DATB. Let the pocket identifier be the SURNAME field (which means that records are compared only if SURNAME agrees on the two records). ROW specifies the row number of the record on the files, and the "..." represents missing data.

Part 2 of Figure 7 shows examples of frequency weights on the WGHT file for the fields SURNAME, MARST and BIRTHYR. (For example, the weight for the surname "Quigley" is "100".) We will be using these weights later to calculate the total weights of links.

Figure 7: Example: two input files and a Weight file

Part 1.-- DATA and DATB file

File	ROW	SURNAME	MARST	BIRTHYR
	1	Barnes	01	1950
D	2	Barnes	..	1950
A	3	Jones	03	....
T	4	Jones	02	1960
A	5	Quigley	03	....
	6	Quigley	02	1960
	1	Barnes	01	1950
	2	Barnes	..	1950
D	3	Barnes	02	1960
A	4	Jones	03	....
T	5	Jones	02	1960
B	6	Jones	..	1950
	7	Jones	02	1960
	8	Quigley	02	1970
	9	Quigley	03	1970

Part 2.-- WGHT file

SURNAME	MARST	BIRTHYR	WEIGHT
Barnes			40
Jones			10
Quigley			100
	01		10
	02		20
	03		30
		1950	10
		1960	20
		1970	30

The table below shows the links we have on the project database LINK file after executing the Compare phase and applying the frequency weights in the WGHT file. The columns in the table are explained below.

Figure 8: Example: the resulting Link file

LINK ROW	DATA ROW	DATB ROW	SURNAME OUTCOME D(-10)	@SURNAME RESULT	MARST OUTCOME D(-20)	BIRTHYR OUTCOME D(-40)	@BIRTHYR RESULT	TOTWGHT	STATUS
1	1	1	A	Barnes	01	A	1950	60	POS
2	2	1	A	Barnes	M	A	1950	50	POS
3	3	4	A	Jones	03	M	....	40	POS
4	4	5	A	Jones	02	A	1960	50	POS
5	4	7	A	Jones	02	A	1960	50	POS
6	5	8	A	Quigley	D	M	.....	80	DEF
7	5	9	A	Quigley	03	M	.....	130	DEF
8	6	8	A	Quigley	02	D	.....	80	DEF
9	6	9	A	Quigley	D	D	.....	40	POS

• THRESH=(40,75) •

Notes:

1. "LINK ROW" identifies the record number of each link. This identifies the link in subsequent reports.
2. "DATA ROW" and "DATB ROW" indicate the File 'A' and File 'B' records that are involved in a link.
3. "SURNAME" and "BIRTHYR" are fields containing the outcomes of comparison. These are "A" (agree), "D" (disagree), "M" (missing on one or both records).
4. For agreement, the "@SURNAME" and "@BIRTHYR" fields contain the result on which the fields agreed.
5. The "MARST" field contains the outcome of the comparison if it is "D" (disagree) or "M" (missing), or the

result on which the fields agreed if the outcome was agreement.

6. For disagreement, the weights are specified under SURNAME, MARST, and BIRTHYR. E.g. for disagreement on BIRTHYR the weight added is "-40".

7. "TOTWGHT" (total weight) is the sum of the relevant agreement and disagreement weights for each link.

8. "STATUS" shows the link status for each link. This is based on the total weight (TOTWGHT) for the link and the current threshold values (THRESH). In this example, the lower threshold is "40", and the upper "75". "POSS" corresponds to 'possible' and "DEF" to 'definite'. (In this example, comparisons resulting in a total weight less than the lower threshold (40) are excluded from further processing.)

For example, for Link 8 we calculate the total weight (TOTWGHT) from the information on the LINK file, and the weights on the WGHT file, as follows:

Figure 9: Example: calculating the weight for Link 8

Comparison	Value	Weight
SURNAME	QUIGLEY	100
MARST	02	20
BIRTHYR	disagree	-40
	TOTWGHT	= 80

The final table below shows the group numbers assigned to the records after grouping. Records with the same group number refer to the same individual. Records having no group number have no matches on the 'other' file. These groups are based on the DATA ROW, DATB ROW, and STATUS values shown on the LINK file.

For example, Group 1 contains three "Barnes" records: A(1), A(2), and B(1), i.e. two File 'A' records have been grouped with one File 'B' record. If our linkage requirement is one-to-one, then this group contains a 'conflict' which will have to be resolved.

Figure 10: Example: group numbers show the linked records

File	ROW	SURNAME	MARST	BIRTHYR	GROUP
	1	Barnes	01	1950	1
	2	Barnes	...	1950	1
DATA	3	Jones	03	....	2
	4	Jones	02	1960	3
	5	Quigley	03	....	4
	6	Quigley	02	1960	4
	1	Barnes	01	1950	1
	2	Barnes	..	1960	...
	3	Barnes	02	1960	...
	4	Jones	03	....	2
DATB	5	Jones	02	1960	3
	6	Jones	..	1960	...
	7	Jones	02	1960	3
	8	Quigley	02	1970	4
	9	Quigley	03	1970	4

7.0 GIRLS TRAINING

As the GIRLS system is relatively complex, we strongly recommend participating in the introductory Seminar, followed by experimenting with an Example Project that has been set up for training purposes.

7.1 GIRLS Seminar

This is a one-day seminar which covers all aspects of the GIRLS system. It is given by the GIRLS system staff on an ad hoc basis. It requires the use of an overhead projector and can be presented at Statistics Canada or elsewhere. This Seminar is a valuable introduction to the system.

7.2 Example Project

This is a miniature GIRLS linkage project with two small files of test data. It consists of a sequence of batch jobs containing examples of the typical use of GIRLS commands. Submitting these jobs one at a time produces a sequence of listings showing the stages by which the records from the two files become linked. You are also encouraged to make a copy of these jobs, change the commands, and then re-submit the jobs to see the effect of your changes. This Example Project is a valuable learning tool.

8.0 HARDWARE AND SOFTWARE REQUIREMENTS

GIRLS requires the following hardware and software:

- IBM 370 compatible hardware with at least two million bytes of storage (real or virtual).
- The OS MVS or MVT operating system.
- The RAPID database management system.
- The IBM PL/1 compiler.
- Direct access storage devices (3330, 3350, 3380 etc.)
- The following are not mandatory but are highly desirable: SAS (Statistical Analysis System) in order to use the Weight Creation function, TSO or ISPF.

NOTES AND REFERENCES

[1] Howe, G.R. and Lindsay, J.(1981). A generalized iterative record linkage system for use in medical follow-up studies. Computers and Biomedical Research, vol 14, 327-340.

[2] Newcombe, H.B. (1967). Record linking: the design of efficient systems for linking records into individual and family histories. American Journal of Human Genetics, vol 19, 335-359.

[3] Fellegi, I.P. and Sunter, A.B. (1969). A theory of record linkage. Journal of the American Statistical Association, vol 64, 1183-1210.

[4] RAPID Database Management System. Informatics Systems Division, Research and General Systems Subdivision, Statistics Canada.