BUREAU OF THE CENSUS
STATISTICAL RESEARCH DIVISION
Statistical Research Report Series
No. RR2001/04

Quality of Very Large Databases

William E. Winkler
Statistical Research Division
Methodology and Standards Directorate
U.S. Bureau of the Census
Washington D.C. 20233

Report Issued: 07/25/2001

Quality of Very Large Databases

William E. Winkler, U.S. Bureau of the Census, william.e.winkler@census.gov

Abstract

Analyses and data mining of large computer files are affected by the quality of the information in the files. For large population registers and for files that are created by merging two or more files, duplicate entries must be identified. Duplicate identification can depend on record linkage software that can deal with name, address, and date-of-birth data containing many typographical errors. Quantitative and qualitative data must be edited to assure that mutually contradictory or missing items are changed automatically and quickly. This paper describes computational methods and software that are suitable for groups of files where individual files contain between 1 million and 4 billion records.

Keywords: record linkage, editing, imputation, data mining

1. INTRODUCTION

There is significant interest in improving the quality of registers, groups of files that might be used in creating data warehouses, merging lists, and identifying duplicates within lists. With the substantial increases in computational power and storage, more groups are able to attempt projects in which single files or groups of files are cleaned to identify and correct erroneous information such as duplicates and contradictory information.

This paper consists of a number of subsections. The second section gives background and covers examples of how duplicates can arise even in well-designed situations. The third section gives background on two methods for improving the quality of data files. The first method is for identifying duplicates. It is based on the Fellegi-Sunter model of record linkage (Fellegi and Sunter 1969). The second set of methods is for assuring the logical consistency of information within a record or group of records. They are based on the Fellegi-Holt model of statistical data editing (Fellegi and Holt 1976). The fourth section covers further examples in which truly enormous files having possibly billions (10^9) of records may be processed. The final section consists of concluding remarks.

2. BACKGROUND AND INTRODUCTORY EXAMPLES

A *duplicate* is a record that cannot be correctly linked with another record to which it corresponds. In a population register, if a record is not given a correct unique identifying number (UID), then it may not be properly connected with other records that are associated with an individual. There are ways to minimize error. The most important is to have a check digit or check digits that are added at the end of the UID. A single check digit can help eliminate 90 percent of erroneous keying and transcription errors and a double check digit can eliminate 99 percent.

If there are no check digits, other quality control methods may not be entirely effective. It is estimated that 2-3 percent of the Social Security Numbers (SSNs) that are used in the California Quarterly Employment Files are in error in any given quarter. Over a period of twenty years, the records with each individual can expect to contain at least two errors where the SSN has been miskeyed or transcribed improperly. The SSN does not have a check digit. For the State of California in the U.S., the twenty-year quarterly employment file contains 1.1 billion records that need to be unduplicated.

The methods of unduplicating the file may involve use of name, date-of-birth information if available, employer, address, and SSN. Each of the identifying fields such as name may contain typographical error. Some of the identifying fields such as employer and address are time dependent. They may not be unique over a period of years.

In some situations, a group may wish to combine multiple files into a large merged database such as a data warehouse. If the files come from a variety of sources, then the files are unlikely to have a UID that allows them to be easily linked. Typically, name and address information may be all that is available for linkages. If a file has been poorly maintained, then the name and address information may be difficult or nearly impossible to use for linkage. Name and a full date of birth are better identifying information than name and address. Even with well maintained business lists, it may be difficult to keep track of the different name variations and different addresses associated with a business over a period of years.

The following table illustrates the difficulty with unduplicating using name information. In line 1, Janice Mary Smith is the current legal married name. The second line, Jan Smith, contains the nickname Jan and might be the form that appears on most lists. In parts of the U.S., it is still possible that many women are listed as in line 3. The form of the name is essentially the husband's name. The fourth line contains two minor typographical errors of the name in line 2. The fifth line is the maiden name that she used prior to being married.

Table 1. Free-form Name Fields in U.S. Lists

1. Janice Mary Smith
2. Jan Smith
3. Mrs. John Robert Smith
4. Jon Smuth
5. Janice Mary Brown

The above names cannot be used for exact character-by-character matching. Name-parsing software (described in the next section) can break a name into components that allow comparison of corresponding components. To facilitate matching, both the married and maiden names need to be maintained in the large administrative list if it is used over a period of years. The Social Security Administration carries the major legal variants of names in its files. Each name is in a separate record that contains the correct SSN. A flag in a separate field indicates what name variant is the currently used version. A name variation such as 3 is essentially unusable. It may be usable if there is auxiliary information that variation 3 corresponds to other variations such as 1. Without additional corroborating information such as address or date-of-birth, it is generally impossible to match on the first name Janice and the last name Smith because they are so common. There are three million individuals with the last name Smith in the U.S. There are 60,000 John Smiths.

The following table indicates variants of addresses. The first three variants all are intended to be actual location where the individual lives at a given point in time. The fourth variant might be a Post Office Box where the individual receives some of her mail. The fifth variant might be the address of an accountant that files the tax forms for the individual. Again, address parsing and standardization software can help with the first three variants of the address. The only way to deal with the last two variants of the address in to carry them as auxiliary information in the address file associated with the individual. Because address information is highly time dependent (in some of the areas of the U.S., twenty percent of the individuals move each year), tracking address information is very difficult.

Table 2.

1. 123 East Main Street
2. 123 E. Main St.
3. 123 E. Main Street, Unit 1
4. P.O. Box 5465
5. 6879 Maple Avenue, Suite 1001

Date-of-birth (dob) information is available in many different forms as illustrated in Table 3. Line 2 is the European convention of day first, whereas line 1 has the U.S. variant with month first. Line 3 is the variant that records the year as two digits, and line 5 is the variant that records the dob in the MMDDYYYY variant in which the year is given four digits. Line 4 has minor typographical errors in both month-of-birth and year-of-birth.

Table 3.

1. January 5, 1960
2. 5 January 1960
3. 01/05/60
4. 01/06/69
5. 01/05/1960

With many U.S. lists, the full date-of-birth is missing with over half of the records. The year-of-birth may all that is available. With a rare U.S. name such as Callahan Zabrinsky, a minor typographical error in the dob field such as given in line 4 of Table 3 may still allow correct matching. With the 60,000 John Smiths, any typographical error in dob is likely to match a John Smith with the incorrect John Smith.

One of the main uses of a large administrative list such as a national health register is in matching it with various hospital, doctor, and regional health records. Each of the lists would need to be statistically edited and imputed to remove or eliminate inconsistent or missing information. For instance, the codes of female for sex and prostrate cancer for disease are inconsistent. Other information in a record might be used to change the sex code to male. More information related to registers in available in Gill (2001).

For various economic analyses, several files might be combined using the name, address, and other information. The merged files might contain quantitative and other data from the source files. Any analyses would need to be corrected for matching error. Some of the quantitative information might require editing and imputing both prior and after matching.

3. **METHODS**

This section describes methods for record linkage and for statistical data editing and imputation. All of the methods have been implemented and used at National Statistical Institutes. With a few exceptions, most of the software can be used on a variety of computer systems.

3.1. Record Linkage Methods

Fellegi and Sunter (1969) introduced a formal mathematical foundation for record linkage. Their model makes rigorous concepts introduced by Newcombe et al. (1960). Two files A and B are matched. The idea is to classify pairs in a product space $\mathbf{A} \times \mathbf{B}$ from two files A and B into M, the set of true matches, and U, the set of true nonmatches. Fellegi and Sunter considered ratios of probabilities of the form:

$$R = P(\gamma \in \Gamma \mid M) / P(\gamma \in \Gamma \mid U) \qquad (1)$$

where γ is an arbitrary agreement pattern in a comparison space Γ. For instance, Γ might consist of eight patterns representing simple agreement or not on the largest name component, street name, and street number. Alternatively, each $\gamma \in \Gamma$ might additionally account for the relative frequency with which specific values of name components such as "Smith", "Zabrinsky", "AAA", and "Capitol" occur. The ratio R or any monotonely increasing function of it such as the natural log is referred to as a matching weight (or score).

The decision rule is given by:

If $R > T_\mu$, then designate pair as a match.
If $T_\lambda \le R \le T_\mu$, then designate pair as a possible match and
hold for clerical review. (2)
If $R < T_\lambda$, then designate pair as a nonmatch.

The cutoff thresholds T_μ and T_λ are determined by a priori error bounds on the rates μ and λ of false matches and false nonmatches, respectively. Rule (2) agrees with intuition. If $\gamma \in \Gamma$ consists primarily of agreements, then it is intuitive that $\gamma \in \Gamma$ would be more likely to occur among matches than nonmatches and ratio (1) would be large. On the other hand, if $\gamma \in \Gamma$ consists primarily of disagreements, then ratio (1) would be small.

Pairs with weights above the upper cut-off are referred to as *designated matches*. Pairs below the lower cut-off are referred to as *designated nonmatches*. The remaining pairs are referred to as *designated potential matches*. The probabilities $P(\gamma\in \Gamma \mid M)$ *and* $P(\gamma\in \Gamma \mid U)$ *are referred to as the m-probability and the u-probability, respectively.* In practice, the probabilities may be difficult to estimate.

The *matching parameters or probabilities* given in the numerator and denominator of (1) can be estimated based on priori experience or via an optimization method such as the EM algorithm (see e.g., Winkler 1995). With very large register files, optimal parameters can be estimated prior to matching and will work well when smaller files are matched against the register (Gill 1999, 2001). If good matching parameters are not available prior to matching, then the parameters can be re-estimated based on a review of the initial matching results.

String comparators are needed because of the large amount of typographical error in files. In some geographic subregions of a major Decennial Census application, as much as 25 percent of first names and 15 percent of last names of records that are true matches contain typographical errors. Typographical error is best dealt with via string comparators that return values between 1 (perfect character-by-character agreement) and 0 (pure disagreement). Table 4 compares the string comparator values returned by the Jaro and Winkler string comparators (see e.g. Jaro 1989, Winkler 1995) with a bigram string comparator that is widely used in computer science. The likelihood ratio in (1) is adjusted for the string comparator values that are strictly between 0 and 1.

Table 4. Comparison of String Comparators Using Last Names, First Names, and Street Names

Two strings		String comparator values		
		Jaro	Winkler	Bigram
SHACKLEFORD	SHACKELFORD	0.970	0.982	0.700
DUNNINGHAM	CUNNIGHAM	0.896	0.896	0.889
NICHLESON	NICHULSON	0.926	0.956	0.625
JONES	JOHNSON	0.790	0.832	0.204
MASSEY	MASSIE	0.889	0.933	0.600
ABROMS	ABRAMS	0.889	0.922	0.600
HARDIN	MARTINEZ	0.000	0.000	0.365
ITMAN	SMITH	0.000	0.000	0.250
JERALDINE	GERALDINE	0.926	0.926	0.875
MARHTA	MARTHA	0.944	0.961	0.400
MICHELLE	MICHAEL	0.869	0.921	0.617
JULIES	JULIUS	0.889	0.933	0.600
TANYA	TONYA	0.867	0.880	0.500
DWAYNE	DUANE	0.822	0.840	0.200
SEAN	SUSAN	0.783	0.805	0.289
JON	JOHN	0.917	0.933	0.408
JON	JAN	0.000	0.000	0.000

Current record linkage software (Winkler 2000) is relatively fast in that it processes approximately 10,000 pairs of records per second. Some commercial software (see e.g. the listing at http://caravel.inria.fr/~galharda/cleaning.html) can be upwards as one third as fast. Most software requires that each input file be sorted by blocking criteria. Blocking criteria are a set of characteristics such as first and last name that every pair must agree exactly (i.e., character-by-character). Sorts that can be prohibitively expensive for a file of one billion records in terms of CPU time (6 days on a fast machine) and disk storage (3.0 terabytes for a 1.0 terabyte file). Software (Yancey and Winkler 2001) that gets around some of the limitations is described in section 4.

To properly match files using name and address information, the components of the names and the components of the addresses must be parsed into components that must be compared. Table 5 illustrates name parsing and standardization. The output is from general business name software (Winkler 1993) that also works well with certain types of person names.

Table 5. Examples of Name Parsing and Standardization

	Standardized
1.	DR John J Smith MD
2.	Smith DRY FRM
3.	Smith & Son ENTP

Parsed

	PRE	FIRST	MID	LAST	POST1	POST2	BUS1	BUS2
1.	DR	John	J	Smith	MD			
2.				Smith			DRY	FRM
3.				Smith		Son	ENTP	

Addresses are considerably more difficult to standardize and parse because they represent far more differing patterns. There are many good commercial address standardization software packages available because of the wide-spread use of mailing lists. Table 6 illustrates examples of address-parsing and standardization subroutines developed by Beck (1994) that is in use at the U.S. Census Bureau.

Table 6. Examples of Address Parsing

	Standardized
1.	16 W Main ST APT 16
2.	RR 2 BX 215
3.	Fuller BLDG SUITE 405
4.	14588 HWY 16 W

Parsed (1)

	Pre2	Hsnm	Stnm	RR	Box
1.	W	16	Main		
2.				2	215
3.					
4.		14588	HWY 16		

Table 6 (continued)

	Parsed (2)				
	Post1	Post2	Unit1	Unit2	Bldg
1.	ST		16		
2.					
3.				405	Fuller
4.		W			

Porter and Winkler (1998) wrote generalized, parameter-driven software that calls the name and address standardization routines.

3.2. Statistical Data Editing and Imputation

A good overview of the principles of Statistical Data Editing is given in Granquist and Kovar (1997). A combination of macro editing can be used to target the largest and most important records for processing manually. The view is further described in De Waal et al. (2000). In some situations, there may be too much data to review clerically. For instance in the 1997 U.S. Census of Manufactures, 100,000 records (4% of 2.5 million records) may contain errors or missing data. Because most of the 100,000 records are associated with small businesses, an automated method can deal with those records. The records of the largest businesses are additionally given a semi-automated clerical review.

The Fellegi and Holt (1976) provided a mathematical model for statistical data editing in which all edits reside in easily maintained tables. In conventional editing, thousands of lines of if-then-else code need to be maintained and debugged. In a Fellegi-Holt system, the code of the main mathematical routines can be easily maintained. It is possible to check the logical consistency of the system prior to the receipt of data. In one pass through the data of an edit-failing record, it is possible to fill in and change values of variables so that the record satisfies all edits. If a complete set of implicit edits can be logically derived prior to editing, then the integer-programming routines that determine the minimal number of fields to change in a record are relatively fast. Implicit edits are those edits that can be logically derived from a set of explicitly defined edits. Generally, it is difficult to derive all implicit edits prior to editing (Garfinkel et al. 1986, Winkler 1997). When most of the implicit edits are available, an efficient way of determining the approximate minimal number of fields to change is described in Winkler and Chen (2001).

In the Fellegi-Holt model, a set of edits is a set of points determined by edit restraints. An edit is failed if a record intersects the set of points. Generally, discrete restraints have been defined for discrete data and linear inequality restraints for continuous data. For continuous x’s,

$\Sigma_i a_{ij} x_j \leq C_j$ for j=1,2,…,n.

For discrete data,

{Age ≤ 15, marital status = Married}.

If a record r falls in the set of restraints defined by the edit, then the record fails the edit. It is intuitive that one field (variable) in a record r must be changed for each failing edit. There is a major difficulty. If fields associated with failing edits are changed, then other edits that did not fail originally will fail. Fellegi and Sunter (1976) showed that implicit edits provide information about edits that do not originally fail but may fail as a record is changed.

The SPEER97 system (Draper and Winkler 1997) for ratio editing and balancing (assuring that items add to totals) is relatively fast (1000 records per second). The DISCRETE edit system (Winkler 1997, Chen 1998) is also fast (1000 records per second). The SPEER97 system requires that most of the implicit edits be computed in advance. The DISCRETE system requires that all of the implicit edits be computed in advance. Both SPEER97 and DISCRETE have modules that assure that imputed records satisfy edits. SPEER97 is known to adequately process relatively large files in which a modest proportion of records have substantial error. As shown by Draper and Winkler (1997), 10,000 (0.4% of 2.5 million) records needed to have 6 or more variables imputed. Of the 10,000, 99.0% were imputed automatically in a manner that assured that the resultant record satisfied edits. Overall, 99.9% of the edit-failing records were imputed in a manner so that the resultant record satisfied edits.

Because computing implicit edits in advance is not always possible, other systems do most of the computation to determine the minimal number of fields to change "on-the-fly". The GEIS system of Statistics Canada (see e.g., Kovar and Winkler 1996) uses a variant of Chernikova's algorithm to perform general linear inequality editing. It processes approximately 10 records per second. A more sophisticated LEO system from Statistics Netherlands (De Waal 2000) simultaneously does linear inequality and general editing. LEO is contained in an edit/imputation system that includes an AutImp module for imputation and an ECS module for finding edit-passing records that are close to imputed records. AutImp does not impute records that satisfy edits. The overall edit/impute system may process as many as 5 records per second. The LEO system is at an early stage of development. Neither GEIS nor LEO/AutImp can assure that records satisfy edits. Both are intended for relatively small situations having 20 or fewer variables in which less than 6 variables need to be changed.

The dramatic reduction in resources by using a Fellegi-Holt type of system is illustrated by Garcia and Thompson (2000). They compared the AGGIES system (Todaro 1999) on a large capital expenditures survey. The edits for the survey are complicated because there are ratio edits and there is some nesting of balance equations. Ten analysts worked up to six months to clerically edit and impute the data. Their changes involved manually making changes and then determining whether the resultant changed record satisfied all edits. By iterating, the analysts were eventually able to produce a record that satisfied all edits. They changed three times as much data as the AGGIES system. The AGGIES system automatically edited and imputed the data in less than 24 hours.

Bankier's Nearest Neighbour Imputation Method (NIM) is an effective alternative edit/imputation methodology. NIM performs well when there are many high quality hot-deck donors (Bankier 1991, Bankier et al. 1997, Bankier 2000). Like pure Fellegi-Holt systems, edits reside in tables that are much more easily maintained than thousands of lines of if-then-else rules. NIM has been used effectively on Canadian and Brazilian censuses. Because NIM is the most thoroughly tested system, the system is likely to be more robust than other systems. It is sufficiently fast to process files with millions of records. The methodology is known to be consistent with the Fellegi-Holt model (Winkler and Chen 2001).

4. **RECORD LINKAGE FOR EXCEPTIONALLY LARGE FILES**

Many individuals believe that identifying duplicates is one of the most difficult of the data quality issues. For a large matching situation such as matching the main Social Security Administration file of 600 million records against the 2000 Decennial Census file of 300 million records, this may entail the detailed comparison of 600 trillion pairs of records. Matching must be done using name, address, and date-of-birth information because the Census file does not contain the Social Security Number. Matching is done on secure administrative-record machines having two additional sets of firewalls inside the main firewalls protecting Census Bureau computers. To match efficiently, the files are matched in a series of blocking passes. During a *blocking pass*, only those pairs agreeing on certain characteristics are considered. For instance, on one blocking pass, only those pairs agreeing on first and last name may be considered. Other characteristics such as dob and address are used to determine whether a pair is a match. On another pass, only those pairs agreeing on date-of-birth may be considered. Prior to each matching pass according to a given blocking criteria, the files must be sorted according to the blocking criteria. Whereas the string comparators are useful once a pair of records has been brought together, they cannot be used for bringing pairs together. Twelve blocking passes have been used in some applications. A sort of a file requires three times the storage of the file being sorted. To sort a 600 million record file of 0.7 terabytes necessitates 2.1 terabytes of storage. The sort can require 3 days on a fast machine. Ten pairs of sorts and associated matching passes can take more than 40 days CPU time and substantial disk storage for intermediate files. The slowest part of the process can sometimes be the amount of skilled programmer intervention that is needed for tracking steps of the processing, backing off intermediate files, and writing auxiliary programs needed for analysis and evaluation.

BigMatch software (Yancey and Winkler 2001) allows the matching of a relatively small file having between 1 million and 100 million records against a large file of 4 billion records. The software allows up to ten simultaneous blocking criteria. For the above situation, the Census file could be divided in three subsets of 100 million records and matched against the Social Security Administration File. For ten blocking criteria, the match would take less than three days (one day for each subset of the Census file). The overall disk space requirement might be as little as 3 terabytes. Very little special programmer intervention would be needed.

BigMatch software begins by storing the smaller file in memory. It proceeds to dynamically build the structures needed for the sort keys, sorts the file by successive sort keys, and stores summary information about the beginning of blocks and the location of individual records within the blocks. Once the data structures are created, matching can proceed. After a record from the large file is input, it is paired with the records in the second file. For each blocking criteria, two files are output. The first file contains the matching weight of the pair, summary information associated with the matching process, and the information from the two pairs that was used in computing the matching weight. The second file contains the record from the larger file that was matched against the smaller file. For each blocking criteria, a special reformatting program creates a printout of pairs by decreasing blocking weight. Another preprocessing programming determines, within each blocking criteria, the sizes of the largest blocks in the smaller file. If blocks are too large, then the blocking criteria can be modified.

A special version of the BigMatch software allows identification of duplicates within a file.

5. **CONCLUDING REMARKS**

With large registers and data warehouses that may contain a billion (10^9) or more records, there is increased need for methods that can identify duplicates within and across files and to statistically edit and impute for missing and contradictory data. This paper describes some of the fastest methods that have been implemented in software.

This paper reports the results of research and analysis undertaken by Census Bureau staff. It has undergone a Census Bureau review more limited in scope than that given to official Census Bureau publications. This report is released to inform interested parties of research and to encourage discussion.

REFERENCES

Alvey, W. and Jamerson, B. (eds.) (1997), *Record Linkage Techniques -- 1997* (Proceedings of An International Record Linkage Workshop and Exposition, March 20-21, 1997, in Arlington VA), Washington, DC: Federal Committee on Statistical Methodology (available at http://www.fcsm.gov).

Bankier, M. (1991), “Alternative Method of Doing Quantitative Variable Imputation,” Statistics Canada Memorandum.

Bankier, M., Houle, A.-M., Luc, M. and Newcombe, P. (1997), “1996 Canadian Census Demographic Variables Imputation,” *American Statistical Association*, *Proceedings of the 1997 Section on Survey Research Methods*, 389-394.

Bankier, M. (2000), “2001 Canadian Census Minimum Change Donor Imputation Methodology,” U.N. Economic Commission for Europe Work Session on Statistical Data Editing, Cardiff, UK, October 2000 (also available at http://www.unece.org/stats/documents/2000.10.sde.htm).

Beck, B. (1994), “Address Parsing Software,” unpublished computer system and documentation, U.S. Bureau of the Census.

Chen, B.-C. (1998), “Set Covering Algorithms in Edit Generation,”*American Statistical Association, Proceedings of the Section on Statistical Computing*, 91-96.

De Waal. (2000), “New Developments in Automatic Edit and Imputation at Statistics Netherlands,” U.N. Economic Commission for Europe Work Session on Statistical Data Editing, Cardiff, UK, October 2000 (also available at http://www.unece.org/stats/documents/2000.10.sde.htm).

Draper, L., and Winkler, W.E. (1997), “Balancing and Ratio Editing with the New SPEER System,” Statistical Research Division Report 97/05 (a shorter version appeared in American Statistical Association, Proceedings of the 1997 Section on Survey Research Methods, pp. 582-587).

Fellegi, I. P. and D. Holt, (1976), "A Systematic Approach to Automatic Edit and Imputation," *Journal of the American Statistical Association*, **71**, 17-35.

Fellegi, I. P., and A. B. Sunter (1969), "A Theory for Record Linkage," *Journal of the American Statistical Association*, **64**, 1183-1210.

Garcia, M. and J. E. Thompson (2000), "Applying the Generalized Edit/Imputation System AGGIES to the Annual Capital Expenditures Survey," International Conference on Establishment Surveys, II, Buffalo, NY, June 2000, to appear in the Conference Proceedings.

Gill, L. (1999), "OX-LINK: The Oxford Medical Record Linkage System," in *Record Linkage Techniques 1997*, Washington, DC: National Academy Press, 15-33.

Gill, L. (2001), *Methods for Automatic Record Matching and Linking and their use in National Statistics*, National Statistics Methodology Series, London: National Statistics.

Granquist, L. and J. G. Kovar, (1997), "Editing of Survey Data: How much is Enough?" in *Survey Measurement in Data Quality*, New York: Wiley, pp. 415-435.

Jaro, M. A. (1989), "Advances in Record-Linkage Methodology as Applied to Matching the 1985 Census of Tampa, Florida," *Journal of the American Statistical Association*, **89**, 414-420.

Kovar, J.G., and Winkler, W.E., (1996), "Editing Economic Data", *American Statistical Association, Proceedings of the Section on Survey Research Methods*, 81-87 (a version is available at http://www.census.gov/srd/www/byyear.html as report rr00/04).

Newcombe, H. B., J. M. Kennedy, S. J. Axford, and A. P. James (1959), "Automatic Linkage of Vital Records," *Science*, **130**, 954-959.

Porter, E. H. and W. E. Winkler (1998), "General Business Name and Address Parsing Software," unpublished computer system and documentation, U.S. Bureau of the Census.

Todaro, T. A. (1999), "Overview and Evaluation of the AGGIES Automated Edit and Imputation System," Room paper presented at the Conference of European Statisticians, 2-4 June, 1999, Rome, Italy.

De Waal, T., F. Van de Pol, and R. Rennsen (2000), "Graphical Macro Editing: Possibilities and Pitfalls" Proceedings of the International Conference on Establishment Surveys, II. 579-588.

Winkler, W. E. (1993), "Business Name Parsing Software," unpublished computer system and documentation, U.S. Bureau of the Census.

Winkler, W. E. (1994), "Advanced Methods for Record Linkage," *Proceedings of the Section on Survey Research Methods*, *American Statistical Association*, 467-472 (longer version report rr94/05 available at http://www.census.gov/srd/www/byyear.html as report rr94/05).

Winkler, W. E. (1995), "Matching and Record Linkage," in B. G. Cox *et al*. (ed.) *Business Survey Methods*, New York: J. Wiley, 355-384.

Winkler, W. E. (1997), "Set Covering and Editing Discrete Data," *Proceedings of the Section on Survey Research Methods*, *American Statistical Association*, 564-569 (longer version report 98/01 available at http://www.census.gov/srd/www/byyear.html).

Winkler, W. E. (1999), "The State of Statistical Data Editing," in *Statistical Data Editing*, Rome: ISTAT, 169-187 (also available at http://www.census.gov/srd/www/byyear.html as report rr99/01).

Winkler, W. E. (2000), Record linkage system with documentation, U.S. Bureau of the Census.

Winkler, W. E. and B.-C. Chen (2001), "Extending the Fellegi-Holt Model of Statistical Data Editing," *Proceedings of the Section on Survey Research Methods*, *American Statistical Association*, to appear.

Yancey, W. E. and W.E. Winkler (2001), "Bigmatch software," unpublished computer system and documentation, U.S. Bureau of the Census.