WHEN, WHY, AND HOW TO DEVELOP WIDELY USED STANDARD SOFTWARE FOR AREA SAMPLING

Thomas Krenzke and James Green, Westat
Thomas Krenzke, 1650 Research Boulevard, Rockville, Maryland 20850

Key Words: Primary sampling unit, stratification, and segment

1. Introduction

An area probability sample survey requires that several complex steps be planned, executed, and delivered in a cost- and time-efficient manner. There continues to be a need for area samples because of their traditionally higher response rates. Due to the ongoing need for area samples and due to its complexities, Westat has created and maintained a proprietary sampling suite of area sampling software.

Westat follows a particular development protocol for standardizing software. This paper will describe that protocol in the context of area sampling software development. Several factors throughout the development process have lead to a widely used product. This paper will include a discussion of "When-to" begin and "Why-to" proceed with statistical software development as well as "How-to" produce a widely used software system. The general underlying methodology for each step of the sampling procedure is discussed and recent users of the software system are mentioned.

2. Background

Area sampling is a multistage sequence of listing and selecting a probability sample of geographic areas. The process begins by forming large geographic areas and selecting a sample of them. The process continues with listing geographic areas within the larger geographic areas selected from the previous stage, selecting a sample, and so on until a manageable geographic area is obtained. Area sampling is necessary for household surveys in order to reduce the cost of listing dwelling units (DUs) and interviewing households or persons. The trade-off to cost reduction is an increase in sampling error due to the clustering of sample units.

The software system that Westat has developed for area sampling handles the following four-stage sampling approach. At the first stage, primary sampling units (PSUs) are formed from counties or groups of counties in the United States subject to some minimum measure of size (e.g., total population of more than 15,000) and travel cost (distance). Once the PSUs are formed, they are stratified and a sample is selected with probability proportionate-to-size (pps). At the second stage, within each sample PSU, segments are formed using census block-level data subject to a minimum measure of size (e.g., 60 DUs). Subsequently, a stratified pps sample of segments is selected.

The third stage begins with a mapping and listing operation. Using the maps as a guide, before listing the dwelling units, the listers go to the location of the sampled segment and count the DUs. If there are more than a specified amount, perhaps 300, then the segment is subdivided into mutually exclusive chunks, of which one chunk is selected with pps. Once the DUs are listed within the segment or chunk, a sample of DUs is selected as the third stage of sampling and the list of sampled DUs is imported into the computer-assisted personal interview (CAPI) system along with address information from the listing procedure.

At the fourth and final stage, all study-eligible people within the DU are listed and one or more persons within a household are selected. More details on the general methodology of each approach are provided in Section 6.

3. The “When-tos”

When is the best time to build a software suite for area sampling? Our development of standardized software has required in general the following:

- The process is understood;
- The process is repeated; and
- The required resources are available.

A well-defined and understood process provides a great basis for developing standardized software. The methodology for area sampling was first developed in the 1940s (Kish, 1965). The understood process of area sampling helped the team to focus on the mechanics (i.e., "how-tos") of developing the software.

A process requires certain volume before developing standardized software can be justified. At the beginning of 2000, Westat had at least three projects needing area samples. This upcoming repeat processing called for a plan to reduce the redundancy of planning, writing specifications, and programming. Most of the thought processing of the task flow, special case handling, parameters, data flow, naming conventions, delivery files, etc., needed to occur just once with a central committed and experienced team. If attempted on a project-by-project basis, the cost of development

would repeat as the task is repeated. In addition, a portion of the process may be missed or the process would not be done as efficiently as if attempted independently.

Staff and funding are obviously critical resources. It helps to have projects lined up to help justify staff use, funds, and to set schedules. The use of project staff during the development phase increases their awareness and education of the methodology and use of the system.

4. The "Why-tos"

Why develop standardized software? In general, we believe that standardized software:

- Reduces total survey costs;
- Reduces total survey error;
- Improves quality through standardization;
- Shortens turnaround time;
- Increases morale by changing the nature of the work; and
- Handles shifts in staff.

Prior to creating the software system, it was necessary to determine if it was worthwhile to invest available resources on a continuous basis. Creating a widely used system is a great effort that involves many people and resources. Westat does not consider a software suite for area sampling as a marketable item, however, there are several reasons for its development and potential for wide use within the company.

Project costs will certainly be reduced due to the investment of undertaking the development of the software suite. The Westat development costs are high, while the client costs are reduced. However, we also improved several other aspects related to the area sampling tasks, as mentioned in the following paragraphs. One major reason for developing the software is to provide a tool for many staff, so that the staff would not be limited or over-burdened to undergo a task such as PSU formation, stratification, etc.

The effort would not begin unless a reduction in survey error could be realized. By building an automated and flexible software system, it is assumed that it would improve the total survey mean square error. For example, forming PSUs had been done manually in the past. When forming PSUs with an automated system, constraints can be relaxed after a first attempt, and then re-processed. With current computer capabilities, the task can be processed under different objectives, constraints, and size measures through the use of specification parameters and the results can be compared to find the best solution to the problem.

Standardization in this case is a mutually agreed upon current best method; at the least its the best method for improving the quality of the sampling task. Standardization of statistical methodology, output, specification format, and computer code were all issues. It was necessary for the team to meet weekly to discuss new ideas and decide whether they should be implemented. Standardized procedures foster efficient documentation as process user's guides serve as the basis for documenting the repeated project work. Another critical benefit to standardization is that it brings staff together and leads to discussions that ultimately at least makes staff aware of the software.

Quality control is a great reason for automating procedures. The area sampling suite has a series of standard checks as a result of the team focusing on process quality. Standard computer output is used to ensure the quality of the resulting product. The checks being automated reduce the burden of having to create a set of checks for each project.

At Westat in general, the statistician writes a spec and the programmer writes the code. Once the software is developed, the programming work is reduced dramatically, which allows staff to focus on the statistical issues that lead to the specification of parameters. The result in the reduction of programmer work is a quick turnaround time for the sampling activities of the project.

The redundancy and tedious nature of the work is also reduced, which allows more focus on more interesting work and may tend to boost morale.

Lastly, since many staff will be familiar with the software, it allows more freedom to shift staff to another project when a staffing shortage occurs. The staff is familiar with the software code from other project work.

5. The "How-tos"

A primary goal is to avoid creating a system that could be used only by a small number of experts familiar with an arcane process. "How-to" create such a system is a challenge and involves open-mindedness and open discussions. The following procedures were used in our effort:

- Discuss the general process and plan carefully;
- Build the team, include users;
- Emphasize flexibility;

- Develop and test;
- Make it user friendly and well documented; and
- Maintain software.

First, the process was discussed in general by a small group of senior staff. We found that a survey of the statistical group was very helpful in initiating the planning process. The survey asked for past experiences in area sampling and for thoughts on usefulness and practicality of certain ideas. This not only was useful in planning, but it began the all-important process of making staff aware that such a product was in the works.

The second step was to build the team. The team consisted of the following:

- Working team;
- Statistical advisory group; and
- Project managers.

The working team attended weekly meetings, carried out the development tasks of writing specifications, and programming. In order for the automated system to be used widely, the software development working team consisted of staff who would become the ultimate users. We also added to the team individuals who had experience in area sampling.

For the area sampling experts not available due to time constraints, we set up a statistical advisory group. The advisory group was available for group meetings from time-to-time or through individual consultation.

The project managers were an invaluable resource. They answered several questions as to practicality issues, corporate goals, future interviewer hiring strategies, PSU size, data necessary for field use, etc. The project managers had experience on area sampling projects.

Third, we realized that it was important to build flexibility in order for the software to be widely used. There are two ways identified:

1. Parameterization (parameter sheets); and
2. Interactive processing.

Parameters were used to allow users the option of processing certain pieces of code written for special-use processing. We took advantage of any existing code written for certain special situations, which allowed for special case handling that may be repeated in the future. Parameters may be constants, objectives, variable names, and filenames that may differ across projects. In addition, parameters may be formulas specified by the user. The specifications written by statisticians include the use of standard parameter sheets that are basically the parameters of a SAS macro run.

Also, it was important to determine the best mode for processing on a step-by-step basis. If it was necessary for the user to interact with the system and data, then a Visual Basic application was developed for that task. These tasks included PSU formation, PSU stratification, chunking, and person sampling. The remaining steps to the process were large processing tasks and were done using SAS macros.

The fourth step was to develop and test the software. New and modified software components were developed from scratch and incurred overhead cost from work done by the working team. However, further testing was conducted through project use. Members of the team associated with projects would report back to the software team on any necessary modifications from their use. It was also helpful that nonteam members were users for writing specifications or programming. This all led to a gradual further development of new versions of individual programs within the software suite.

The fifth step was to make the system user-friendly. We consider the following items as being essential for this purpose:

- Centralization;
- Documentation;
- Training; and
- Dissemination.

The centralization of statisticians, programmers, and even network directory access has proven to be highly conducive to the usefulness of our software systems. Through a standard documentation notebook/user's guides with examples, staff lunch talks, short distances to meet/discuss certain aspects of the system, the potential set of users will have a convenient means to getting their tasks completed. Staff training sessions will provide some more education and awareness of the system and its uses. The dissemination of the software is as easy as granting access to a centralized network directory.

The sixth step was to maintain the software. Over time software can become out-of-date due to new data available, and new methods and computer capabilities. To ensure that the software meets the needs of users, new versions of each software component within the system can be created, documentation can periodically be updated, and notes to the user group can be sent.

6. General Methodology

Figure 1 shows a flowchart of the area sampling process. Part of what makes this a system is that it is set up to flow data through each component smoothly without preparation of the files prior to processing a particular step.

The top row of Figure 1 shows the components of the PSU processing. The new components of PSU processing include WesPSU, WesStrat, and PostPSU. The process begins by forming PSUs in WesPSU. PSUs are formed from adjacent counties subject to a particular objective function and a number of constraints. The primary objective is to minimize the travel distance to reduce costs of data collection. However, a second objective is to form PSUs such that the between PSU variance is minimized. The PSUs are formed using the minimax or minimin methodology noted in Lindo (1998). Green, Chowdhury, and Krenzke (2002) provide a detailed description of WesPSU. WesPSU is a Visual Basic application.

Next, the PSUs are stratified in WesStrat to group PSUs close in characteristics to reduce the sampling variability between PSUs within each stratum. The stratification method follows a nested hierarchical structure, where stratifiers are categorized. The basic flow is that once the PSU file is read in and parameters specified, the number of strata assigned to each major stratum is allocated. The first stratifier, second stratifier, etc., is specified and then several stratification scenarios are presented. The 'best' scenario can be selected by the user based on a selected evaluation criteria, including a between PSU measure and equal-sized strata measure. WesStrat is also a Visual Basic application. We considered the U.S. Census Bureau's PSU stratification software, which is based on a variation of the Friedman-Rubin approach (Jewitt and Judkins, 1988). The algorithm used by the Census Bureau for clustering based on multivariate continuous variables exceeded our needs. However, we also favored the nested hierarchical approach due to its clearly defined strata.

Next, PSUs are selected using WesSamp if one PSU is selected per stratum, or Durbin's Method (Durbin, 1967) if two PSUs are selected per stratum. WesSamp is a general purpose sampling macro that is used for many types of sampling tasks. We have also written a SAS macro for Durbin's Method. After PSUs are selected, a SAS macro (PostPSU) assigns IDs and PSU names.

The middle row of Figure 1 shows the components of the segment processing. Enhancements were made to two components (WesBlock and WesSeg), while two new pieces were added (PostSeg and ListPrep). The Census 2000 SF1 block-level files are prepared in WesBlock for forming segments. WesBlock assembles required information for segment formation in WesSeg. WesBlock will handle blocks with zero population and zero housing units based on a set of specified parameters. In addition, WesBlock can attach information from the Census 2000 SF3 block-group files. The macro also subsets the census blocks to the selected PSUs.

The WesSeg macro combines contiguous or near-contiguous census blocks until a minimum measure of size is met. Data are aggregated from the block- to segment-level. Parameters are available to request formation within tract boundaries, or within block group boundaries. WesSeg now handles the new census block numbering system. WesSamp is then processed to select the segments.

Post-segment selection processing using the SAS macro PostSeg calculates the expected, minimum, and maximum number of DUs in each segment for guiding the listing operation. Segment IDs are assigned to the sample segments. PostSeg also produces the expected sample yield for a number of stages in data collection (occupation rate, eligibility rate, screener response rate, interview response rate, etc.). Summary tables are produced with weighted estimates in the sample for checking purposes. Lastly, PostSeg provides the data necessary for preparing files for mapping, chunking, and listing.

The SAS macro ListPrep is a simple program that produces files in optional formats for use in mapping, listing, and chunking. All the above programs were written in order to maintain a smooth data flow between system functions.

The bottom row of Figure 1 addresses the dwelling unit and person sampling stages. The WesList operation is primarily a segment management tool in Visual Basic to help record information from sampling chunks and listing segments or chunks. Within WesList, field managers can select chunks through pps sampling and capture the sampling data that are necessary for selecting the DUs. WesList has recently been expanded to include segment management operations, providing the facility to assign and re-assign listers to regions, enter dates that listing sheets have been received, and also generate reports providing listing and chunking frequencies.

Once the DUs have been counted for each sampled segment, the WesSamp macro is processed to select the DUs for the study. The sample DUs are identified and then the DU address information is keypunched. The sampling and location information comprises the survey control file, which is entered into the CAPI system. Specifications are written to CAPI

programmers who implement the sampling algorithm into the CAPI program for person sampling.

7. Recent Software System Uses

In this section, recent users of the redesign system are mentioned. The software was first tested on the Early Childhood Longitudinal Study - Birth Cohort (ECLS-B) project, conducted for the National Center for Education Statistics (NCES). Another early usage of WesPSU was for the National Head Start Impact Study (sponsor is Agency for Children, Youth, and Families in the U.S. Department of Health and Human Services), when the software was still under development. The input file consisted of counties containing Head Start programs. PSUs were formed with a minimum of eight Head Start programs. The counties in each PSU were not necessarily contiguous in this case.

Each of the following adult literacy studies, conducted for NCES, has used all components of the software suite for area sampling: National Assessment of Adult Literacy, State Assessment of Adult Literacy, and the Adult Literacy and Lifeskills for the Department of Education.

For the 2003 Commercial Building Energy Consumption Survey for the Department of Energy, a special measure of size was created for counties based on commercial activity. Project staff are forming PSUs using WesPSU within states and metropolitan statistical areas in order to combine counties to reach the minimum measure of size for PSUs while restricting driving distance (in progress).

For the National Study of Health and Activity, within major stratum groups defined by Census Division and MSA status, substrata were formed using county percentages of race, ethnicity, and poverty. WesStrat was used to assist in this process. All components of the software suite were used for area sampling tasks.

WesPSU was used to create PSUs for the Department of Transportation Commercial Truck Survey. The input file was all counties in 12 states containing Interstates or Limited Access Highways. The county measure of size was miles of Interstates/Limited Access Highways. WesPSU was also used recently for forming PSUs for the Survey of Youth in Residential Placement.

WesPSU and WesStrat were used recently for the National Health and Nutrition Examination Study, conducted for the National Center for Health Statistics. This software significantly reduced the amount of time that would have been required to create strata (with the aim of having approximately equal-sized strata) and to combine PSUs with measure of size (MOS) below the minimum MOS. Additionally, the results obtained using the software were more optimal than what could have been obtained through a manual process.

WesPSU was also used to form the PSUs for two special studies (Writing On-line and Oral Reading) as part of the National Assessment for Educational Progress.

8. Summary

Westat's area sampling software was initially developed in the mid-1990s. Since then, the system has been modified on a continuous basis. Starting in 2000, new components to the process have been implemented, including:

- PSU Formation;
- PSU Stratification;
- Post-PSU Processing;
- Post-segment processing; and
- Preparations for listing, chunking and mapping.

In addition, enhancements were made to the WesBlock and WesSeg macros, as well as to the WesList software component.

The cost and usefulness of the software system benefited from a carefully planned and efficiently executed development process involving considerations of “When-to”, “Why-to”, and “How-to” develop the software system for wide use.

9. Acknowledgments

The working team for the area sampling software redesign consisted of Laura Alvarez-Rojas, Jim Bethel, John Burke, Sadeq Chowdhury, Jim Green, Tom Hankins, Brice Hart, Wen-Chau Haung, Andrew Heller, Katie Hubbell, John Edmonds, Leyla Mohadjer, David Morganstein, Debby Vivari, Tom Krenzke, William Wall, and Ian Whitlock.

The statistical advisory group included David Judkins, Graham Kalton, Keith Rust, and Joe Waksberg.

The operations management consultants were Renee Slobasky, Martha Berlin, and Pat Montalvan.

In addition, the existing software from the mid-1990s was a large effort that included many who wrote specs and code, and all those involved in the planning and development teams throughout the years.

We also thank the U.S. Census Bureau staff, led by the late Chip Alexander, for their correspondence and discussion with regard to their PSU stratification system.

10. References

Durbin, J. (1967). Design of multistage surveys for the estimation of sampling errors. *Applied Statist.* 16, pp. 152-164.

Green, J., Chowdhury, S., and Krenzke, T. (2002). WesPSU—Development and demonstration of primary sampling unit (PSU) formation software. *Proceedings of the Section of Survey Research Methods of the American Statistical Association.*

Jewett, R. and Judkins, D. (1988). Multivariate stratification with size constraints. *SIAM Journal of Scientific Statistical Computing*, (9)6, pp. 1091-1097.

Kish, L. (1965). *Survey sampling*. New York: John Wiley & Sons.

Schrage, L. (1998). 2nd ed. *Optimization modeling with lingo*. Chicago, IL: Lindo Systems.

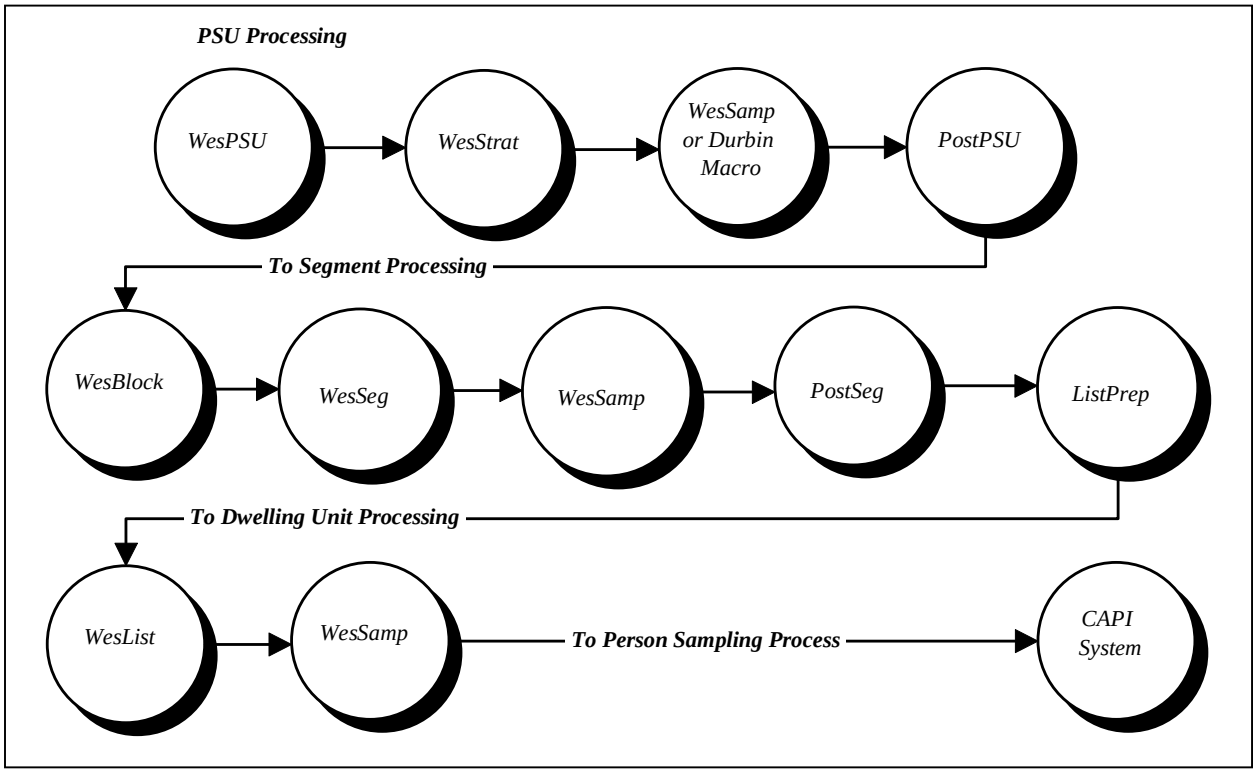

Figure 1. General flow of software suite for area sampling