Lessons learned on the development of the Unified Enterprise Survey

Rock Lemay, Statistics Canada

1. Introduction

The Unified Enterprise Survey program incorporates several annual business surveys into a single framework, using questionnaires with a consistent look, structure and content.

The 2002 version of the Unified Enterprise Survey encompasses 41 surveys, collecting standard data points from both large and small businesses across multiple industries, resulting in more coherent and accurate statistics on the economy, particularly at the provincial/territorial and industry levels.

The collection process is done at Statistic Canada's headquarters with a mailout/mail-back approach of over 40,000 questionnaires, using a multi phase mail-out system based on the respondent's fiscal year end, to a maximum of 3 phases.

During the course of this paper, I will take you through the whole process of our system and in particular, the reasoning behind each software and design approach that was adopted. The goal of the system design was to reduce the amount of time required for maintenance and development, but to also give the collection team tools to improve their day-to-day operations, such as analytical reports, to help in their decisions.

A plan to solely use Blaise for all of these processes and controls would not allow us to grow with the increasing demand to streamline the collection operations and the complexity of the questionnaires being designed to reduce response burden.

For UES, we have determined that Blaise's biggest strength is the processing of records at the micro level. It allows for extremely efficient ways of keeping track of edits behavior. However, the speed at which it processes vast amounts of records has pushed us to investigate the use of other software to complement the whole UES collection system process.

The system can be broken down into 6 different areas, each with its own software approach.

2. Mailout System

The first step in our mailout/mail-back approach to survey taking is the preparation of questionnaires, address inscriptions, and control mechanisms. The approach taken for this particular process was to use SAS/AF, due to its analytical power of large files and the speed at which it can process data. The system generates mailout files that are sent for addressograph printing and questionnaire stuffing into envelopes. It also facilitates separating the files and sending them to a repository area awaiting the load of these cases into various Blaise applications.

3. Blaise applications, Frame Feedback and Central Document Control (CDC)

Each of the 41 questionnaires is a distinct Blaise application, but all records are loaded into a Central Document Control and a Frame Feedback Blaise database. Even with the consistent framework approach to the questionnaire, there are enough differences to require us to build a different Blaise application for each questionnaire. Each of them has on average 100 edits, with approximately 60 of them being generic to all.

The design allows us flexibility during the collection period to change or fix individual applications without disrupting others. However this is not true for the CDC and the Frame Feedback, since all applications contain external calls to these. Should an enhancement be required for either of the two main components after deployment or even during testing, it requires us to recompile all 41 applications. Another drawback is the fact that we have 41 call schedulers (CATI Specs), which makes it difficult for the Collection area to manage, since some of their interviewers work on more than one questionnaire.

One of the approaches we are considering is the use of SAS as a CDC, being the only point of entry for all cases. The Collection area has requested a greater control for case management and follow-up. Because of the mailout/mail-back approach, two kinds of follow-up are required; one for non-response and the other for edits both requiring different types of staff. Using SAS/AF with a built-in selection grid would allow the Collection Managers a greater flexibility in targeting cases depending on the time of collection (i.e.: beginning of collection - non-response follow-up). Using this strategy would necessitate the use of the BCP to allow for the transfer of data between the Blaise application and the SAS dataset. With the BCP, any changes to the CDC would not require major recompiles of the 41 Blaise applications. In using SAS, the creation of MIS reports would be sped up, compare to the share mode utilized in Blaise.

4. Analysis tools

One of Blaise's features is the creation of a file with the extension .BTH, which adds an observation every time someone exits a case in true CATI mode. The BTH and edit evaluation features present in Blaise lend themselves well to the survey data processing and historical case access recording. There is a need to analyze this type of data to improve questionnaire design and reduce collection cost.

Analyzing the BTH files has allowed us to measure the efforts done by the collection area. This however was not fully representative of the total effort since it only recorded entries accessed by the DEP. Entry into the CDC is via Maniplus and therefore not recorded. We have created Manipula processes that are called on exit of cases, regardless of their access point. These processes create entries in a pseudo-BTH file. Using SAS, we gather all BTH information for a particular questionnaire and are able to prepare daily progress reports in real-time. This type of data is also extremely useful in debugging, since it demonstrates the full history of case access. In addition, we have started to compile and analyze using SAS, the Blaise Audit Trail data to provide additional information on the collection process.

5. Data Capture

During the initial years of development, application design was based on a CATI approach, with skip patterns and "**.Show".** After a few collection cycles, we

noticed that only 15% of all data was captured during CATI sessions. The remainder was from questionnaires, which were later keyed using a non-interactive edit mode. The application was a reflection of the questionnaire and the interviewer would have to go through all cells to typically capture a small number of items. To bring efficiency to this costly process, a string key approach was developed using Blaise, which we named Quick Data Entry (QDE). Only cells with data are captured and subsequently re-integrated into the application. A Quality Control System was later deployed to ensure reasonable accuracy during capture, using Visual Basic to create reports and manage QC parameters such as 100% or batch verification.

This data capture process led us to develop QDE Edit Reports. The Collection areas needed a way to quantify the amount of edits that were failing from the initial capture or respondent data, prior to any intervention from the interviewer. After re-integration of the QDE, which is done in overnight batches, the **CheckRule** is applied and an extraction of the database is done with all the failed edits. To bring meaning to this information, SAS with HTML outputs are used to compile the data into reports for further analysis. This is done for every application, which enables the Subject Matter Areas to review their questionnaire design with the information provided.

The same process can be used to measure the quality of any data needing re-integration in Blaise, such as electronic data reporting or key from image.

Problems with data retention occurs during the re-integration process when data on the questionnaire does not follow the skip patterns set out in the CATI Blaise application. The solution is to show all cells and eliminate the skip pattern flow. We have lost some efficiency in navigating through the application, but have ensured data retention regardless of what respondents fill out.

6. Follow-Up

To optimize resource utilization during collection, a better targeting of units had to be employed. A **Dynamic Score Function** was developed. The logic behind this approach is to stop follow-up when your weighted response rate has been reached and transfer resources in strata where more data is required to produce adequate estimates. Using SAS, a calculation on key variables from clean records is used to recalculate the priority of each case. This is then re-integrated into the Blaise application. The CATI specifications give priorities to those cases and ignore others where response rate targets have been met.

7. Outputs

The Estimation and Imputation system for the UES is programmed in SAS. A need to output the data into SAS datasets was required to allow quick and efficient process. All data outputs are done from Blaise to ASCII to SAS. Programming the output for all 41 applications would be quite time consuming. By utilizing a standard output specification format in Excel and Visual Basic, we developed a **Code Generator** to produce the Manipula and SAS programs. The same Code Generator is also used to create the QDE re-integration programs.

8. Conclusion

As you can see from the steps above, the development of our business model in a modular approach has given us the flexibility to grow with the ever-changing requirements from all stakeholders. Since a large percentage of monetary resources of a survey are allocated to the collection area, a need to understand how and where they are spent is of the utmost importance. Not only do we need to analyze this data, but we must also be in a position to re-tool the systems from the findings of that analysis.

We have determined that Blaise and SAS provide us great tools to ensure micro level data management as well as versatile macro data manipulation.

Further research is being done to improve the UES collection process, such as customized questionnaires generated in XML using previous year's respondent data. Other components requiring data summation are being redesigned with the approaches learned from the UES, such as using SAS for dataset manipulation and Blaise for record level validation and verification.

Blaise will remain as a major player in our collection systems, while we continue to integrate components utilizing whatever software is deemed the most appropriate. This process will continue to evolve, due to the nature of technology and the ever-changing needs of all players involved in Survey taking.