Contents

1. **Blaise and Methods of Documentation**
   - Michigan Questionnaire Documentation System (MQDS): A User’s Perspective ........................................4
   - Blaise and XML: Experiences Of An Outsourcing Project .................................................................5

2. **Audit Trails**
   - Using audit trails to monitor respondent behaviour in an Audio-CASI questionnaire .......................8
   - A solution with the aim of an optimal evaluation of the time of interview under CAPI: the statistical exploitation and analysis of the audit trail file (.ADT) with the software S.A.S ........................................9

3. **CATI & CATI Mgt**
   - Computer Assisted Dialing: What will it do for you? .................................................................12
   - CATI Management: Behind the Scenes ...................................................................................13

4. **BCP**
   - Experiences Using an Event History Calendar ................................................................................16
   - Retrospective Data Collection ......................................................................................................17
   - Lifecycle Processes to Insure Quality of Blaise Interview Data ......................................................18
   - Navigating BCP with .NET ...........................................................................................................19
   - The ScriptWriter Tool: .................................................................................................................20

5. **Using Blaise 4.8**
   - Methods for Simultaneous Meta and Data Manipulation in Blaise .............................................22
   - Encrypting Blaise Data on Network Servers ...............................................................................23
   - Generic Data Storage with Blaise 4.8 Datalink ............................................................................24

6. **Testing Blaise Surveys**
   - Case Scenario Library for Testing ...............................................................................................26
   - Blaise Testing ...............................................................................................................................27
   - CTT: CAI Testing Tool ................................................................................................................28
7. **Blaise and the Internet**  
Analyses of web survey data .......................................................... 30  
Web Application Stress Testing and Blaise IS .................................. 31

8. **Blaise Questionnaire Specification, Design, & Implementation**  
Blaise Source Code Editing System .................................................. 34  
Survey Specifications Management at Statistics Canada .................. 35  
Challenges in Converting the National Crime Victimization Survey to Blaise ... 36  
Designing and Developing Person-based and Topic-Based Data Collection Instruments ........ 37  
Coping with people who just won’t stay put: The Use of Blaise in Longitudinal Panel Surveys ...... 38  
Authoring Blaise questionnaires – A task for the survey specialist or an IT programmer? .... 39  
The ONS Integrated Household Survey: The next installment ................ 40  
BLAISE in Macedonia ......................................................................... 41  
Using Blaise 4.7 for data entry, checking, error reporting and transforming for further processing ...... 42  
Questionnaire Specification Database for Blaise Surveys ..................... 43  
Exploration of Blaise Instrument Generation from Metadata .................. 44

9. **Blaise Programming**  
Dynamic List Building Using Blaise .................................................. 46  
The use of mobile devices to carry out Blaise surveys at ONS ................. 47  
Using the Blaise Alien Router For Audio-Recording of In-Person Interviews .......... 48

10. **Poster Sessions**  
Miscellaneous Examples of Programming in Blaise and Manipula .................. 50  
Designing an e-form to collect survey data .......................................... 51  
New CAI-operation at Statistics Norway ........................................... 52  
Deploying a C# .NET Object onto a Laptop to be used with a Blaise 4.7 Data Collection Instrument .. 53  
Conversion of Blaise Database to Relational Databases .......................... 54
Abstracts by Category

1. Blaise and Methods of Documentation
Michigan Questionnaire Documentation System (MQDS): A User’s Perspective

Heidi Guyer and Gina-Qian Cheung
The University of Michigan

The Michigan Questionnaire Documentation System (MQDS) is a survey documentation tool developed by the Survey Research Center at the University of Michigan based on the Blaise® System. MQDS can be used to produce questionnaire documentation and codebooks for instruments programmed in Blaise. The system is web-based allowing for the inclusion of linked data such as logic routing or skips patterns, respondent booklet text, question objectives, and other such ancillary information. Other features of the MQDS include the capability to produce documentation in multiple languages, a utility for producing a SAS dataset from Blaise data, and a utility for merging documentation from large or complex data sets. The compare feature of the system allows for the comparison of Blaise datamodels. This feature can be used to compare different data models within a study, as well as to compare similar questionnaires or sections between studies. This presentation will provide an overview and demonstration of the various MQDS utilities.
Blaise and XML: Experiences Of An Outsourcing Project

Leif Bochis Madsen
Statistics Denmark

Xml is a standard for the representation of tree structures with decorated nodes and today it has become a common format for exchanging information in a platform-independent way.

Because of their hierarchical nature Blaise data and meta data are quite easy to convert into xml, however, various demands may influence the way this should be done – among them conformance to standards requirements and – of course – the particular application.

Statistics Denmark has recently been through a process of outsourcing telephone as well as personal interviewing for the Labour Force Survey while post processing of forms including coding still should be carried out in our office.

Among the tasks were definitions of a data exchange format using xml for the exchange of questionnaire data and development of tools for automatic generation of e.g. xml schemas to support validation of xml documents and for conversion between Blaise and xml.

The paper summarizes some experiences and challenges from this project.
2. Audit Trails
Using audit trails to monitor respondent behaviour in an Audio-CASI questionnaire

Olivier Bart
INSEE, France

From November 2005 to January 2006, Insee has conducted the Événements de Vie et Santé (EVS) survey under CAPI. After a face-to-face interview dealing with relationships between tragic events of life (death or suicides of close relations, violence…) and health, respondents had to answer a brief ACASI questionnaire gathering the most sensitive questions: sexual activity, alcohol and drug use.

We have collected adt information for around 9000 complete surveys, which enabled us to consider a statistical study on this data. As it was the first time we had this information available on an ACASI questionnaire, we chose to focus on the ACASI part to monitor respondent behaviour facing this tool, and try to improve data collection methodology.

This paper will explain how we processed the adt files and present the main results we obtained on the following issues:

- What is the average time spent by the respondent before giving an answer?
- Do respondents replay questions? Do they reread or correct their answers?
- Who are the people who give up the ACASI questionnaire before the end?

All aspects will be discussed from the respondent’s point of view (age, sex, nationality, academic level) and from the questionnaire’s point of view (open/closed questions, questions about dates/facts/feelings …).
A solution with the aim of an optimal evaluation of the time of interview under CAPI: the statistical exploitation and analysis of the audit trail file (.ADT) with the software S.A.S

Georges Bourdallé
CAPI Application Manager,
I.N.S.E.E (France)
georges.bourdalle@insee.fr

Objective: To allow the designer of household survey to optimize at the same time the rates of payment of the household interviews under CAPI and the household number to be investigated.

Description: In the preparatory phase of implementation of a household survey, the designer of it is confronted with two ambitions, at the same time to reward most exactly possible the work of the household survey interviewers and to investigate most possible households. The payment of questionnaires represents to him only 60 % of the budget of a household survey. It is thus necessary to optimize the budget assigned to a household survey.

For that purpose, several levels of payment or rate of payment are set up. Of the weakest in the most mattering there is a rate for the waste (impossible to join, long-term absentee, refusal,…), a rate for the partially performed questionnaires and a rate for the totally performed questionnaires. This last rate being the most expensive because supposed to reflect a relatively important duration of interview, it is thus extremely important that it reflects most exactly possible the reality.

The time of interview is estimated during the field test.

The exploitation of the audit trail files (.ADT) of the household survey interviewers at the end of the test data collection allows us to create various indicators indispensable in the statistical analysis with the aim of the evaluation of a duration of interview either for the totality of the questions of a given survey, or for a set of questions or for only one question. These indicators are to know the identifier of the household, the information relative to the household interviewer (investigator’s number, the regional direction of the INSEE where is connected the investigator), the number of passage (opening and lock) in the questionnaire, set of questions or for only one question, the time for every passage, the time of the longest passage and the total time defined as the sum of the time of all the passages.

The statistical analysis allows us to have quickly an estimation of the duration of interview for a complete questionnaire independently of an effect investigator. According to the budget, we can so simulate various durations of interview by the deletion of questions.

So the most just possible estimation of the duration of interview can be quickly made.

The statistical analysis of audit trail files (.ADT) and shaping of files results are made with the software S.A.S (Statistical Analysis System) by the implementation of macro S.A.S.
3. CATI & CATI Mgt
Computer Assisted Dialing: What will it do for you?

Dan Bernard
Marketing Systems Group

Computer assisted dialing is realizing wider and wider use, driven mostly by realization of significant productivity gains. Recently, dialers are being used to facilitate the use of new technological advancements such as VoIP for virtual call center implementation and audio recording for significant gains in quality control and training. Implementation of dialer usage by survey research organizations such as NORC has made major in-roads into the acceptance of the technology in the survey research field.

Marketing Systems Group, creators of GENESYS Sampling Systems in the US, will present what the environment is today for automated dialing - and where it will likely go tomorrow. Benefits, in addition to productivity, will be explored. Emerging technology which will improve productivity, yet minimize call abandonment will be addressed.

Learner Outcomes:

- Differentiating between auto, power and predictive dialing.
- How does automated dialing work:
  - Dialer interaction with the BLAISE system.
  - The role of statistics.
  - What the respondent hears.
- New technologies being introduced to:
  - Improve productivity, but minimize call abandonment.
  - Facilitate the building of virtual call centers using VoIP.
  - Significant gains in quality control and training using audio recording.
- What are some of the productivity gains that can be expected.
- What are some the experiences organizations have had.
  - How automated dialing improves job quality for your interviewing staff.
  - Improving project quality.
  - Do the interviewers feel controlled by the computer.
- What is the industry adoption curve of predictive dialing.
  - How prevalent is it?
  - Will I have to do it too?
- Justifying the cost of the dialer.
  - Are dialers expensive?
  - Do I need a large phone room to make automated dialing pay off?
  - The dialer replacing the need for a PBX.
CATI Management: Behind the Scenes

David Dybicki
The University of Michigan

Blaise CATI out of the box has a series of features designed to maximize calling efficiencies. However, the efficiency comes at a trade off with maximizing response rates. When using the call scheduler to satisfy a more balanced mix between efficiencies and response rates we must run a series of custom processes in order to get the desired effect that optimizes the autodelivery of sample for both dimensions. In this paper I will describe techniques we use programmatically using Blaise manipula scripts in conjunction with Windows task Manager to maximize our sample delivery throughput.
4. BCP
Experiences Using an Event History Calendar in the Panel Study of Income Dynamics

April Beaulé, Eva Leissou, and Youhong Liu
The University of Michigan

The Panel Study of Income Dynamics is a nationally representative longitudinal study of approximately 8000 U.S. families. The Board of Directors of the PSID continually search for ways to innovate the data collection instrument. For the 36th wave of the PSID, a major new component was added. A childhood health calendar was programmed to collect information on the most common childhood conditions including asthma, diabetes and allergies. Since this childhood health component is retrospective focusing on the time from the respondent’s birth to age 17, the design team felt that using a calendar module would help trigger the respondent’s memory. The PSID had prior experience using an electronic calendar for employment data outside of the main Blaise interview module. This separate employment calendar presented us with many obstacles, the most significant of which was the processing of separate file structures: Blaise and Access databases. In order to alleviate some of these processing issues, the decision was made to incorporate a Visual Basic calendar that would be called from the Blaise DEP. The data from the VB calendar application was written directly back to the Blaise bdb using a Dynamic Link Library thus avoiding multiple datasets.

In addition to data structure and extraction issues, the calendar mode versus the standard question list presented us with a number of challenges during interviewer training. This paper will discuss the challenges that we faced when fielding this calendar including data structure, extraction, and interviewer training. This paper will also present preliminary results of calendar data and how it will be used to generate final output variables for the PSID public use files.
This paper will outline the selection, design, and use of a CAPI event history calendar (EHC) to collect retrospective life course data. In contrast to traditional linear questioning models (Q-lists), the event history approach encourages the sequential and parallel retrieval of information from autobiographical memory, and has been found to produce better-quality retrospective reports (Belli et al 2001). This method was employed to collect detailed information about key events that had occurred in the lives of approximately 8,850 participants from the English Longitudinal Study of Ageing (ELSA). We aimed to enhance our understanding of how early life events influence the circumstances of older people by asking participants to recall events that had happened prior to joining the ELSA study in 2002. Key dimensions covered were relationships and fertility; housing and mobility; jobs and earnings; and health.

The following aspects of the event history calendar will be described in the paper:
(a) Advantages of an event history approach over traditional questioning methods in terms of improving recall and increasing user-friendliness
(b) Outline of the time units and domains covered in the calendar
(c) Issues relating to the design of the calendar - e.g. flexibility of the interview, changing responses during the interview, displaying events on the calendar
(d) The role of Blaise and Visual Basic as the calendar interface
(e) Implications for interviewer training
(f) Coding and editing of data
(g) Application to other large-scale social surveys

---

Efficient, accurate and timely post-collection processing of interview data is critical in both cross sectional and longitudinal survey environments. Through the use of the Blaise API a comprehensive approach has been implemented for the post-collection processing of Blaise interview data. This paper discusses the processing approaches used to manage the dynamics of changing data models; post-collection comment review and edit processes; codebook preparation; tracking the status of edit completion; and handling data complexities within the system. The experiences implementing a lifecycle approach for COTS Blaise studies across multiple survey environments within an organization are discussed including implications for backend systems architecture and functionality.
Navigating BCP with .NET

Peter Sparks
The University of Michigan

The Blaise Component Pack is a gateway to expanding the capabilities of Blaise by working with metadata, data, and other parts of the Blaise system. However, finding just the right information can be difficult, and the current documented examples are in Visual Basic 6 and C++. This paper looks at the different components available, how navigate them from the various entry points using the .Net 2005 programming environment in both VB and C# and BCP 2.0, and some tips to make coding easier. Equivalent examples will be given in both Visual Basic.Net and C#.Net.
The ScriptWriter Tool: An Application for Interviewer Training Script Development

Youhong Liu
The University of Michigan

Interviewer training scripts play an integral role in Computer Assisted Interviewing (CAI) studies. They are used in interviewer trainings to provide interviewers with practice through a Blaise questionnaire, while trainers highlight important features. Scripts are also used to: assess interviewer questionnaire administration; ensure the quality of data; and offer a means to standardize training elements. Standardization is crucial to the proper training and certification of interviewers, especially across training teams with multiple training sessions.

Script development can often present a challenge for those involved. The difficulty of their production is primarily technical. Traditionally, script writers have hand-entered information from the Blaise screen. This consumes significant time and staff resources, while increasing the likelihood of inaccurate and inconsistent entry by multiple writers. The need for a more efficient, automated approach to script development led to the creation of an application, called Scriptwriter.

The ScriptWriter Tool is a Visual Basic .NET program. It allows users to enter survey responses within a stand-a-lone version of the Blaise instrument, while entering training annotations in parallel. Through an interface in ScriptWriter, users have the option to edit training annotations, specify output settings, and produce script output in .HTML format. This paper will outline the development and subsequent use of the ScriptWriter Tool, and discuss the challenges in creating a user-friendly application with the flexibility to serve the needs of many projects and diverse Blaise applications.
5. Using Blaise 4.8
Methods for Simultaneous Meta and Data Manipulation in Blaise

Marien Lina
Statistics Netherlands

This paper describes a number of practical aspects for different methods in Blaise to combine the manipulation of meta and data.

Firstly, Blaise III introduced Cameleon to produce meta descriptions. It addresses the meta only and it is mostly combined with Manipula setups when data should be addressed.

Secondly, Blaise Windows introduced the Blaise Component Pack (BCP), which can address the meta and the data simultaneously in one Visual Basic application.

Thirdly, related to this, Blaise 4.8 possesses two new features: In Manipula you can read and manipulate meta and data simultaneously, and: the data entry program (DEP) can be combined with Manipula setups more easily than before. A data form edited in the DEP can be manipulated in manipula while the record is still in memory and not saved in the data file. When calling manipula the edited form is available in the manipula process and values can be returned from Manipula to the DEP.

The paper describes a number of specific practical survey tasks and concludes with a number of advantages and disadvantages of the three methods for performing these tasks.
Encrypting Blaise Data on Network Servers

John Mamer
Mathematica Policy Research, Inc.

Given the United States Federal Government’s increased focus on security, government agencies are beginning to require all confidential data be encrypted both during transmission and while “at rest”. The focus for the encryption of data at rest has been on laptops, but some agencies are applying the encryption requirement to data on network servers behind corporate firewalls, implying that the all Blaise data reside on network servers in encrypted format. This paper reports on conclusions drawn from Mathematica’s investigation and testing of a number of encryption mechanisms for Blaise datasets on network servers.
Generic Data Storage with Blaise 4.8 Datalink

Arno Rouschen
Statistics Netherlands
aron@cbs.nl

Because database software and servers have become more and more powerful over the years, many organizations tend to store their data in relational database management systems. This has several advantages, among others that data is not stored locally on disk, can be secured and is not proprietary and thus accessible from all kinds of third party software.

From Blaise 4.6 / BCP 2.0 on it is possible to store Blaise questionnaire data in relational database management systems (RDBMS), like Oracle and Microsoft SQL Server. We name this concept Blaise Datalink. Datalink uses so-called Blaise OleDb Interface (BOI) files. Within Blaise you can use such a BOI file in the same way as a native Blaise database file. Blaise OleDb wizards are available to create a BOI file and the corresponding database tables for your Blaise questionnaire. The BOI file contains connection parameters and information about available tables and how to retrieve field data from these tables.

Although it is already a nice feature that you are able to store Blaise questionnaire data in an RDBMS, we had some ideas and also had some requests from clients to make the storage more generic; it would especially be nice if we could share database tables between questionnaires. In previous Blaise Datalink aware versions, BOI files were tailor-made for a particular survey and each Blaise questionnaire had its own set of database tables.

Blaise 4.8 Datalink has been extended and can define, next to the custom-made BOI files, so called generic BOI files. Generic BOI files use a common primary key in order to store the data in a generic way, but this works completely transparent to the user; you just need to define, as usual, primary and secondary keys in your Blaise data model. By using generic BOI files, the number of needed Blaise tables in your RDBMS decreases, because tables are shared and less tables means less database administration. Also it will be easier, from an architectonic view, to integrate Blaise data in your production environment, because all Blaise questionnaire data lives only in a few tables in your RDBMS, and the data itself can be accessed in a generic way, for example by your case management system or other parts of your software environment.

Blaise 4.8 BOI surveys are registered on your database server. This gives us the possibility to do some survey administration on the server, for example to insert a new survey, delete an obsolete one, show the state of a particular survey and to query Blaise questionnaire data. Blaise 4.8 ships with a new OleDb client tool especially developed in order to support these features.

The presentation demonstrates how to set up generic tables with help from Blaise 4.8 BOI files, explains a little bit of the inner workings of Blaise 4.8 Datalink and also shows you how to administer the server with the new Blaise 4.8 OleDb client tool.
6. Testing Blaise Surveys
The Health and Retirement Study (HRS) is a major national longitudinal survey on the health and economics of aging and retirement. It consists of a complex multi-sectional 1-2 hour-long interview of 22,000+ participants, conducted every two years using CAI technology. It utilizes preload of more than 350 variables from previous waves, and incorporates seven mode and language combinations within one datamodel.

The scope of the HRS instrument exacerbates the common difficulties inherent in accurately programming and testing a CAI application. HRS endeavored to develop a testing system to reduce these difficulties by conceiving of and building a group of utilities called the HRS Case Manager - “HRS-CM.” Together, these programs make use of stored keystroke files that are produced as a by-product of Blaise test interviews.

A new feature of HRS-CM is the scenario system. By using stored ADT files and preload values, the scenario system allows testers to quickly recall and re-apply keystrokes to ensure that a programming error was fixed and additional errors were not introduced. This greatly reduces re-testing time. Moreover, the scenario system has proven to be effective for reviewing real interviews exactly as they were taken and has substantially reduced the time needed to investigate problems reported from the field. The system’s success is reflected in a reduced number of reported errors from the field, in comparison to the numbers reported in previous waves.

There were some significant challenges to overcome in order to build an effective scenario system. Discussion will cover issues relating to data storage format of both preload and ADT/ADK files, re-creating the environment for test replication, and re-applying ADK/ADT files after paths have been changed/corrected within the application. Other important features include capabilities for saving, documenting and recalling scenarios, as well as a utility that evaluates the scenarios for instrument coverage.

The success HRS has had in using HRS-CM could be applicable to other studies. It is hoped that a discussion of this project will generate interest in this aspect of instrument development.
Blaise Testing

Rebecca Lui, Daniel Collison, and Margaret Tang
Statistics Canada

Blaise software was introduced at Statistics Canada in 1999. Since then, Blaise has been the designated software used to develop CATI and CAPI survey collection instruments. Currently, there are more than 200 Statistics Canada surveys in production using Blaise 4.6.

Deciding whether to upgrade to a newer version or a newer build of Blaise is always a difficult decision for the development managers. Once the survey is in production, the development managers are reluctant to change to another version of Blaise for future cycles. When upgrading to a new version, significant testing efforts are often needed to ensure that all the functions are working properly. Although a newer version of Blaise may offer many attractive features, the cost and risk for the survey associated with making the change, may not seem worthwhile.

The newest version of Blaise (4.8) offers many new features to improve application connectivity and scalability. In order to test these new features in a cost effective way, Statistics Canada has created a project to develop a testing strategy to certify the new version. This paper describes the complexity of the testing issue and the approach taken to establish a reusable testing process.
Without adequate tools for systematic testing of the instruments, a research study may experience production delays, interviewer frustration with instruments that don’t function properly, or collection of imperfect or incomplete data that fails to meet the needs of the research project.

To increase the over-all quality of Computer-Aided Instruments through standardized testing procedures, reduce the cost of testing, increase access to information concerning CAI development and software de-bugging through preset and ad hoc performance metrics reports, the Computer-Aided Instrument Testing Tool was developed.

The Computer-Aided Instrument Testing Tool (CTT) is comprehensive tool set that is used for testing and reviewing survey instruments and reporting on the testing process. The CTT application was developed to facilitate the testing of Blaise survey instruments by a variety of local and remote users. The CTT application automatically captures and consolidates testing comments, provides bug reports, assigns items to application developers and records testing outcomes.

This paper will discuss the major components of the CTT application which will include the Preload builder, Instrument testing, bug recording, random case generator and reporting on testing notes. The CTT Administration system will be reviewed as well.
7. Blaise and the Internet
Analyzes of web survey data

Vesa Kuusela
Social Survey Unit
Statistics Finland

The data capture process in a web survey differs from that of a postal survey and an interviewer administered survey. The differences have give rise to new types of errors in data and therefore they should be filtered out or otherwise be taken into account in the data analysis of the survey.

In an interviewer-administered survey the interviewer controls what is entered in the form and he or she also is able to assess the feasibility and reliability of the answers. Usually the obtained data will be ready for analysis after minor checking. Also unit non-response is coded adequately and may be processes using available methods. Item non-response remains (usually) in a moderate level.

In a postal survey, like in a web survey, the data capture process cannot be controlled. However, in a postal survey, the forms have to be returned to the institution or company conducting the survey and that requires some effort. On the other hand, sending a static web questionnaire is usually a very simple task and when using an interactive web questionnaire the data is stored all the time. Often in a postal survey the returned forms will be checked before data entry. The validity of the data is assessed (and invalid data filtered) in checking of the forms and partly also during data entry. Therefore, in a postal survey the data to be analysed is composed of answers, which can be considered given seriously. In a web survey one cannot be sure of that. For example, some respondents may have only浏览 through the questionnaire and entered the answers only to reach the end of the form.

In a web survey, only partly filled forms, interruptions, is a new phenomenon which practically nonexistent in interviewer administered surveys and postal surveys, because interviews are interrupted very seldom and only partly filled mail questionnaires are not sent. Especially, in and interactive web questionnaire interruptions may become a problem.

The paper discusses different erroneous ways to fill in a web form and in which ways those could be discovered. Results of a data cleaning analysis of a web survey data will also be presented. The web survey was published in a newspaper with a general invitation and without an identification the respondents. The analysis shows that 10% - 25% of returned forms could (should) be disregarded, depending on the criteria for exclusion.
Web Application Stress Testing and Blaise IS

Jim O’Reilly
Westat

An important consideration for web surveys is the scalability of an application: how many simultaneous users can the server configuration support before response time degrades significantly. The issue may be more relevant for Blaise IS in that it enables implementation of large complex web surveys more so than many other systems. This paper discusses techniques and approaches for implementing stress testing Blaise IS applications using the Microsoft Windows Application Stress (WAS) software. Topics covered include:

- Adapting IS datamodels and IS supporting elements for stress testing
- WAS procedures and metrics
- Integrating IS journal data and IS database information with WAS metrics
- Stress test reporting
- Approaches to automating WAS stress testing
- Considerations for single server and multiple server architecture implementations
8. Blaise Questionnaire Specification, Design, & Implementation
Blaise Source Code Editing System

Sheila Deskins and Danilo Gutierrez
The University of Michigan

The paper discusses a Blaise Source Code Editing System that was designed and developed by Health and Retirement Study (HRS) programmers. It covers the six major design components of a source editing system, as well as specifics relating to application design, development, and testing plans that allowed the creation of the current working system.

The six major design components are: 1) a file reader/parser to process the Blaise source code files, 2) a maker of Blaise statement targets, 3) a translator to change electronic update information into a format the system can process, 4) a merger that combines the update information and the Blaise source code, 5) a file writer to produce the revised files, and 6) an interface to encompass the whole system.

The Blaise Source Code Editing System or “Source Editor” is used to make several hundreds of updates to the Blaise Source files (.bla/.inc) automatically. In the past the process of updating information from electronic review systems had been incorporated into Blaise code by hand, which was a labor-intensive, tedious and error prone process.

The Source Editor has the capacity to handle Blaise applications of approximately 3,500 defined fields, six defined languages, 175,600 lines of source code, 518 procedures, and 344 blocks. Although the Source Editor has options that are HRS specific, it can be run using non-HRS Blaise Applications. It can accommodate changes in the language order, number of languages, and the number of language-related statement targets.

Along with handling updates from review systems, the Source Editor has the capability to run bulk updates. The Source Editor updates the .bla and .inc files automatically, while preserving comments, white space, and the .bla and .inc file structure.
Survey Specifications Management at Statistics Canada

Pamela Ford
Operations Research and Development Division,
Statistics Canada, Ottawa, Canada

The Framework Systems Section of the Operations Research and Development Division (ORDD-FSS) is responsible for designing and developing the collection technology framework and architecture for survey collection operations within Statistics Canada. Collection technology architecture provides a framework within which the survey life cycle is managed. This includes supporting the specification, development, testing and production deployment of survey instruments, the management of collection operations during collection, the delivery of survey data from collection to down-stream subject matter processing systems and the post-collection analysis of operational data.

One of a number of initiatives identified as priorities for ORDD-FSS is the development of a specifications repository that will encourage the reuse of survey specifications and associated instrument components when common concepts are used within multiple surveys.

This paper focuses on the proposed approach for specifications development and management and on the development and usage of a specifications repository within this process.
Challenges in Converting the National Crime Victimization Survey to Blaise

Charlie Carter, Roberto Picha, Dan Moshinsky, Alexis Arrington, Michael Mangiapane
Technologies Management Office
U.S. Census Bureau
(301) 763-3473; (301) 763-7730

Introduction

The National Crime and Victimization Survey (NCVS) is a major national longitudinal survey with a sample size of about 56,000 households. This survey provides personal victimization and property crime data on types and incidences of crime; monetary losses and physical injuries due to crime; along with characteristics of the victims and offenders. Although NCVS is a household based survey, it collects crime data for every household member who is age 12 and older. The NCVS was converted from a Pen and Paper Interview (PAPI)/Computer Assisted Telephone Interview (CATI) CASES interview to a Computer Assisted Personal Interview (CAPI)/CATI Blaise instrument in July 2006.

Survey design

The NCVS has a set of screener questions that determine the type and number of times a specific crime has occurred within a six month reference period for each eligible household member that leads to a set of questions that provide more detail information for every person who has reported a crime. To avoid the problem of reporting the same crime multiple times between household respondents, the survey also has an un-duplication component that compares the crimes reported and flags duplicates for deletion. The final component of the survey is an editing and coding operation that is integrated within the regular NCVS instrument and is administered after data collection has been completed.

Implementation Challenges

We had serious performance issues because of the initial requirements to collect up to 30 crimes for each eligible member of the household – with a maximum limit of 30 people per household. This requirement results in a large number of arrayed blocks being generated (30 * 30 = 900).

Another implementation challenge had to do with developing an approach for comparing crimes between members of the household that contained enough detail for the interviewer to determine if the crime is a duplicate of one that has already been reported.

The third challenge with the NCVS instrument was integrating an “Editing and Coding” instrument that only allows a certain set of questions to be eligible for review, editing, or coding.

Conclusion

This paper will discuss our three major challenges and the approach we used to address them. It will also discuss the structure of the instrument and the difficulties inherent in converting a survey of the size and complexity of NCVS from CASES to BAISE.
Designing and Developing Person-based and Topic-Based Data Collection Instruments

Emily A. Johnson and Thomas C. Melaney
US Census Bureau, USA

Data collection operations at the U.S. Census Bureau often require gathering similar information from multiple persons in the same housing unit. There are reasons for collecting this data in a “person-based” format; that is, asking the entire series of questions about each individual, one-by-one, in the unit. Likewise, there are other reasons for using a “topic-based” technique to collect data, where each question is asked, one-by-one, of each household member.

This paper will examine a case study of the American Community Survey Content Test. Two instruments were programmed using Blaise 4.6 software, one of them using a person-based approach and the other using a topic-based approach in order to determine which method would be more effective. We will briefly review the background for considering the alternate approaches, discuss in some detail the design and development issues involved in creating the alternate data collection instruments, and consider what enhancements could be included in the Blaise software that would enrich the process.
Coping with people who just won’t stay put: The Use of Blaise in Longitudinal Panel Surveys

Colin Setchfield
ONS, UK


ONS has much previous experience of conducting longitudinal studies (particularly with its Labour Force Survey) and has developed a method within Blaise for transferring data from one wave to the next. GHS, however, presented fresh challenges.

• Persons moving out of the original household to new addresses were to be kept in sample but removed from the original case; a new case (with its own unique serial number) needed to be created for them, which would read data from the original case as appropriate only for those movers.
• Persons moving into the original household were to become part of the sample and to be added to the existing case.
• Arrays within the Blaise datamodel were required to take account of these changes within household with redundant elements removed for more effective database storage.

This seemed surmountable only by going against the ONS Blaise-programming mantra of “Keep it simple.”

This paper will look at the key issues in designing this Blaise instrument, including solutions that were found to collapse/expand arrays to match the new number of household members, and to force interviewers to open up and automatically populate new households for movers before signing off the original household.

The GHS is one of the surveys to be incorporated into the new Integrated Household Survey. ONS has also developed a new longitudinal study – the Wealth and Assets Survey – with two-yearly waves between interviews. The GHS changes have therefore become part of our standard for programming longitudinal surveys.

The paper will conclude by considering the impact of the complicatedness of the code generated: in terms of control, maintenance, and the ONS model of researchers programming questionnaires.
Authoring Blaise questionnaires - A task for the survey specialist or an IT programmer?

Rebecca Gatward
Office for National Statistics (ONS)

All organisations approach the development of Blaise questionnaires using their own unique model. Their approaches, however, can be broadly divided into two models. In the first, a dedicated team of Blaise programmers develops the questionnaire based on specifications produced by the survey specialist. In the second, survey specialists develop the Blaise questionnaire themselves.

Most Blaise-user organisations have been using the package for between five and ten years, and therefore now would be an appropriate time to review these two approaches.

This paper will provide a review of current Blaise authoring practices across organisations - based on information obtained from Blaise users within the organisations. Each user will be asked to provide detail about their current practice, including, a description of the model used to develop Blaise questionnaires, how the current approach has evolved, an estimate of development time and if possible an indication of the scope and level of accuracy of their Blaise questionnaires.

The discussion will include a review of models: looking at the advantages and disadvantages of each, exploring why approaches differ between organisations, and drawing some conclusions about the efficiency of the various models.

Rebecca Gatward is a Principal Survey Researcher at the Office for National Statistics (ONS)
At the 2004 and 2006 Blaise conferences the concept of and ONS’s plans for an Integrated Household Survey (IHS) were outlined. This paper will provide an update on the progress made on the development of the IHS.

Initial field test trials of the IHS were based on the use of a single Blaise instrument, with elements of the IHS component surveys grouped together under an additional level of hierarchy. In the most recent development work carried out, four separate Blaise datamodels were used; one for each individual survey ‘stream’.

This paper will describe the rationale for adopting this particular approach in the recent development work and the alternatives considered in the process. It will examine some of the problems and challenges encountered. These will include both practical and methodological issues such as serial numbers, version control and the consideration of the different survey reference week across the IHS component surveys.

This paper will consider the lessons learnt from a systems test of the IHS conducted in February 2007 and present results from a further pilot scheduled for June/July 2007.

One of the goals of the IHS is to develop a survey system capable of accommodating other surveys at a later stage. The paper will report on an additional survey, the English Housing Survey (EHS), which will be part of the IHS from April 2008, and the provisions made for the inclusion of the EHS in the IHS design.
The Labour Force Survey is one of the most significant statistical actions that is conducted by the State Statistical Office. It is one of the most valuable sources of information regarding the events in the labour market. The data of this survey are collected through interviews by trained interviewers, who collect the information in a traditional manner, which means on printed forms (questionnaires). Within the framework of this survey the State Statistical Office has also conducted Test Survey on Households through laptop, but after the performed data processing it was decided to continue with collection of data through printed questionnaires.

The entry application for data processing, in accordance with the available technology and equipment in the State Statistical Office, is in BLAISE software with online controls. At the beginning of the preparation of the application, professional assistance was given by the Danish-Finnish Consortium.

Each year modifications were made in the questionnaire and in the entry application with a purpose to improve the quality of the survey material. This also understands training of external workers (persons to make entries) by the Office.

The further data processing is in SAS software and finally the data are put in DB2-6000 basic data base.

The Labour Force Survey is the first BLAISE application in the Office. This application was for a long time one of a kind, but finally the advantages that this software provides were understood. So, today in the same software we have also other surveys. We expect their number to increase.
I like to tell you about our experience in using BLAISE survey processing system. We use BLAISE for data entry and control about a year. During that period all new data entry programs were made in BLAISE. Some of the data entry applications for the ongoing surveys were rebuild in BLAISE too. Finding it very quick and convenient especially for large databases our intention is to use it for major part of the surveys we provide.

The things are set like this: The application for certain survey in BLAISE is settled on server in special folder for that survey. Operations performed are: data entry with online control, batch control and reporting errors using manipula, corrections and transforming data using cameleon, in formats suitable for further processing. Regarding that our database is DB2 for all surveys, and that so far we don’t use direct communication between DB2 and BLAISE, the data are always transformed in text format so that it would be appropriate for further manipulation or maintain. Sometimes data is transformed in SAS format according to the needs. All these operations are performed thru interface built in Visual Basic 6 and are partly automatically. But we have vision and work on it to make environment, which will provide possibilities for whole processing data to be more automated.

Contact: orhideja.krstanova@stat.gov.mk
Questionnaire Specification Database for Blaise Surveys

Lilia Filipenko, Joe Nofziger, and Valentina Grouverman

RTI International

CAI development is a team effort that presents many challenges. When a group of specification writers, translators, programmers, and testers are working in parallel, specifications quickly get out of sync with program code. The Questionnaire Specifications Database (QSD) facilitates all facets of the CAI development lifecycle and streamlines the process of creating Blaise instruments, especially those with a second language and/or ACASI.

More than a common code generator, QSD allows spec writers and translators to make iterative changes at any point in development, while maintaining the integrity of the instrument. Changes to wording, question fills, and response types are inserted directly into the Blaise instruments. QSD’s user-friendly specifications are complete with response options, fills, and question text in two languages. Since these specs and the current Blaise code are both generated from QSD, they are always in sync. Extensive support for ACASI includes script generation, audio review, and synchronization of code with audio files. Issue tracking and change logging are built in features.

QSD is a management tool that tracks all aspects of questionnaire development. It has reduced Blaise instrument development time at RTI. This paper will describe the functionality of QSD in some detail.
Exploration of Blaise Instrument Generation from Metadata

Fred Wensing and Juanita Pettit
Australian Bureau of Statistics

Blaise instruments consist of a structured set of fields, along with routing, edits and layout instructions. Instruments are often scripted from specifications, which define the instrument content and behaviour. If sufficient rigour were applied to the definition of this metadata, would it be possible to build a system that could generate the source code for Blaise instruments?

This paper explores the issues associated with the generation of Blaise instrument source code from metadata. It commences with a detailed analysis of the elements which make up an electronic questionnaire, and leads into the development of a prototype facility which is expected to deliver Blaise instruments from metadata.

While some level of instrument generation appears possible, it can only be done if the metadata is comprehensive enough. This is particularly so if the code generation facility is to support different modes of collection (eg. personal interviewing or self-completed form). The generation of instrument code will also be assisted by having standardised instrument designs and some preprogrammed questionnaire elements. A good, flexible metadata management facility also seems to be fundamental to the success of any code generation system. These and other issues will be discussed in the paper.
9. Blaise Programming
Data collection instruments frequently ask redundant questions that collect data containing the same or similar information. For instance, enumerating questions might collect addresses for multiple people where each person has the same address, or might have a different or alternate address. This paper addresses the problems and issues of building a dynamic list of respondent answers and using that list as a pick list for answers to future questions in the survey for the current administration session.

To reduce both interviewer error entering the data as well as respondent burden repeating the same answers over and over again, Blaise can be programmed to build a list of suitable answers maintained within each case. For subsequent times through these questions, the instrument will display a list of the previously entered answers. If the next respondent answer is on the list the interviewer can simply pick the correct entry. If the respondent offers a “new” answer, that answer will be entered and added to the list. That answer will become available the next time a question is asked using that set of categories. This paper will discuss the programming necessary to build such lists.
The use of mobile devices to carry out Blaise surveys at ONS

Tim Burrell
ONS, UK

The Office for National Statistics uses the Blaise DEP loaded onto laptops for all social data collection. This includes the International Passenger Survey (IPS) where data are collected from passengers entering and exiting the country on paper forms, then keyed into a Blaise questionnaire to be transmitted to head office.

For more than 10 years, ONS has been open to the idea of switching to an immediate data entry tool on the IPS, such as a hand-held or mobile device. In recent years there have been advances in hardware, plus new developments in Blaise, which have now led us to investigate the use of Blaise on hand-held devices.

This paper will report on trials running Blaise on handhelds. It will analyse flexible Blaise is to transfer from a laptop to a hand-held device and any interviewer training issues.
Using the Blaise Alien Router For Audio-Recording of In-Person Interviews

M. Rita Thissen, Sridevi Sattaluri, and Lilia Filippenko
RTI International

The Blaise alien router is a useful technique for managing calls to external software. In this paper, we describe use of an alien router for activation of audio-recording software for in-person surveys.

The performance of field staff and the effectiveness of item wording are notoriously hard to monitor during production field interviews. Traditional methods of live observation can bias results though the presence of the observer, and verification call-backs may not provide enough detail. For computer-assisted face-to-face surveys, digital audio recording offers an efficient alternative which allows evaluation of interviewer and item performance during data collection.

We describe a practical technique in which Blaise instruments can be enabled for computer audio-recorded interviewing (CARI). This approach comprises an external file for specifying items to be recorded, an alien router which activates the laptop’s sound-recording system, and block-level activation of the router within a Blaise instrument. Ideally, a system for audio recording should require little programming, execute undetectably, provide a flexible choice of which items to record, allow respondent consent to be given or revoked at any time and store recordings in clearly identified files. This system offers all of those advantages. By invoking the router on specific blocks or on the entire instrument, a minor programming effort can turn a standard implementation into a CARI-enabled questionnaire.
10. Poster Sessions
Miscellaneous Examples of Programming in Blaise and Manipula

Rob Groeneveld
Statistics Netherlands

1. Anonymizing A Blaise Database

Starting with a Blaise database with identifiable records, we would like to get rid of the primary key, which may be some identifying variable like the name of a company or a registration code for a person or company. It is desirable not only to strip the identifier, but also to sort the records in the database in random order so as to nullify the association between the order of the records and the order of the identifying field.

2. Printing Open Fields In Field Order In A Blaise Database

In a Blaise database obtained as a result of a survey, the task is to print the Open Fields in field order, i.e., firstly the answers from all records in the first open field, then the answers from all records in the next open field, and so on. A procedure can do part of the job.

3. Programming Partially Filled, Nested Arrays In A Survey

The usual way of programming an array in a Blaise survey is to first ask for the number of repetitions, e.g., the number of people in the household, followed by a FOR loop to put a set of questions to each member of the household. Another way is asking, after each set of questions, if there is a further set of questions to be asked. Moreover, within each set of questions one or more sets of questions must be asked, i.e., the arrays are nested. It is shown how a datamodel like this can be applied to an inventory of the knowledge of people in a department, both in terms of the areas of expertise and the length of the time spent in specific jobs.
Designing an e-form to collect survey data

David Kinnear
ONS, UK

ONS currently uses Blaise as a collection tool for its social surveys, via telephone, face-to-face and self-completion. It has also used web collection for the internal ONS Staff Perception Survey. However, the introduction of Blaise IS and BASIL allows ONS the possibility of collecting business survey data over the internet. At present, ONS uses TDE (Telephone Data Entry) and imaging software to capture data from paper forms. Using Blaise Internet could provide ONS the opportunity to offer businesses another option for returning their data.

This poster session will show the results of our project to design an e-form based on existing ONS paper forms. It will include versions created using Blaise 4.7 IS and BASIL from Blaise 4.8. We will highlight any issues we encountered whilst creating the e-forms, and show images of the completed forms that were uploaded to the ONS intranet site.
New CAI-operation at Statistics Norway

Hilde Degerdal and Jan Haslund
Statistics Norway

In 1999 Statistics Norway build a new system for Computer assisted interviewing out by the field interviewers. The system has been a good frame for the work of our work. But after a while you always find things that should have been improved. The main weakness of our system is the communication. It was build as a dial-back RAS connection with a reverse proxy server, polled from inside. Also firewalls and encryption are in use. It is a rather complicated solution, but it had to be this way to meet the claim from Data Inspectorate and the protector of Privacy in Norway. The result is that the rate of unsuccessful connect attempt is too high.

In our present system each respondent is sent to an interviewer as a package. If the interviewer of one reason or another is unable to commit the interview, he has to connect again and return it to the central database. Then, in the office, a person has to find a new interviewer for the respondent and prepare it for a new transfer. This work is quite time consuming. Another problem is that when the respondents are out by the interviewers the office has little control of the collection progress by each interviewer.

Today new and better possibilities for online operations are available. We have a project going on to make a solution where all telephone interviewing shall be done being online with the database in the head office. We plan to use the Blaise CATI call scheduler to handle the workload and appointments. The system shall support different levels of dedication of respondents to the interviewers. It might be like we want one-to-one relationship between respondent and interviewer, or to one group of interviewers, in some surveys or in some phases of the data collection. In other situations all respondents shall be available to all interviewers like in ordinary CATI.

For the communication system we think of using broadband for the interviewers where that will be possible. Then we will use IP-telephone for the interview.

Approximately 80% of the interviews conducted by our field interviewers are done by phone. The coverage of wireless connections in Norway is not good enough to have online systems for face to face interview. So for the face to face interviews we have to find a solution of transfer of the respondents to the interviewers’ laptops.

In the early phase where we are now, we are not able to tell more about the system.

According to the plans we will have a clear sketch of the solution in September. But we will be in a phase in which it will be fruitful to discuss the solutions with other people, and perhaps get some new input. Therefore we want a presentation with discussions and perhaps possibilities to show some of the ideas of the new CAI-operation in Statistics Norway, called SIV.
Deploying a C# .NET Object onto a Laptop to be used with a Blaise 4.7 Data Collection Instrument

Mecene Desormice, Daniel Moshinsky, and Thomas C. Melaney
US Census Bureau, USA

The U.S. Census Bureau is currently designing and developing a new annual survey, the Dynamics of Economic Well-Being System (DEWS), that is intended to provide statistical information on household income and participation in income assistance programs and serve as a successor to the Survey of Income and Program Participation (SIPP). In order to facilitate accurate recall for DEWS, sub-annual data for the survey will be collected with the help of an Event History Calendar (EHC) which, as the name implies, will ask respondents to record key events into a calendar. The functionality of the EHC is being designed and developed in a C# program which will be called from Blaise.

The paper will identify the steps necessary to deploy a Blaise survey with a .NET DLL component onto a system that lacks a full MS Visual Studio installation. An alternative successful strategy will also be discussed. Implications for deploying this kind of instrument onto hundreds of laptops configured for use by enumerators in the field will be discussed as well.
Conversion of Blaise Database to Relational Databases

Yogeeta Purohit
Latha Srinivasamohan
William Dyer
U.S. Census Bureau

Blaise database (.bdb) files can be converted to relational databases such as Access, SQL Server, Oracle, etc.

This paper demonstrates the use of the BOI (Blaise OLE DB Interface) file, in Blaise 4.7, along with Maniplus to convert the Blaise databases into an Access relational database format. This task was earlier accomplished using BCP (Blaise Component Pack) and API library.

Blaise version 4.7 provided a toolbox to automate the transformation process by creating the BOI files. Since this release there have been further enhancements to improve the transformation process. With the deployment of the Blaise IS for use in Internet data collection we see opportunities to apply this tool in the client-server environment.

This paper will discuss some of the advantages of exporting data directly to another database and data manipulations that can be done in Microsoft Access (for example) to support end user applications. Understanding the relationship between the Blaise Datamodel and the relational database format, often familiar to the client, is essential to integrating the data collection effort with their end user systems. Some of the benefits of relational databases will be reviewed including: the normalization of the data structure; Blaise for data collection; relevance of post interview data processing and data security.
Index

A
Arrington, Alexis 36
Ash, Rhonda 26
Australian Bureau of Statistics 44

B
Bart, Olivier 8
Beaulé, April 16
Bernard, Dan 12
Bibb, Barbara S. 46
Bochis Madsen, Leif 5
Bourdallé, Georges 9
Burrell, Tim 47

C
Carter, Charlie 36
Cheshire, Hayley 17
Cheung, Gina-Qian 4
Clark, Pat 18
Collison, Daniel 27
Cox, Kate 17

D
Dascola, Mary 28
Degerdal, Hilde 52
Deleva, Mira 41
Descans, Sheila 34
Desormice, Mecene 53
Dybicki, David 13
Dyer, William 54

F
Fiacco, Alessio 40
Filipenko, Lilia 43, 48
Ford, Pamela 35

G
Gatward, Rebecca 39
Gowen, Linda 18
Groeneveld, Rob 50
Grouverman, Valentina 43
Gutierrez, Danilo 34
Guyer, Heidi 4

H
Hacker, Elizabeth 17
Haslund, Jan 52

I
I.N.S.E.E (France) 8, 9

J
Johnson, Emily A. 37

K
Kinnear, David 51
Kuusela, Vesa 30

L
Leissou, Eva 16
Lessof, Carli 17
Lina, Marien 22
Liu, Youhong 16, 20
Lui, Rebecca 27

M
Mamer, John 23
Mangiapane, Michael 36
Marketing Systems Group 12
Mathematica Policy Research 23
Melaney, Thomas C. 37, 53
Miceli, Colin 17
Moshinsky, Dan 36
Moshinsky, Daniel 53

N
National Centre for Social Research (NatCen) 17
Nofziger, Joe 43

O
O’Reilly, Jim 31
Office for National Statistics (ONS) 38, 39, 40, 47, 51
Orhideja, Krstanova 42
Ostergren, Jason 26

P
Pettit, Juanita 44
Pacha, Roberto 36
Purohit, Yogeeta 54

R
Rodriguez, Gilbert 46
Rouschen, Arno 24
RTI International 43, 46, 48

S
Sattaluri, Sridevi 48
Setchfield, Colin 38
Shepherd, Jane 18
Smith, Jeff 28
Sparks, Peter 19
Srinivasamohan, Latha 54
SSO R. Macedonija 42
State Statistical Office of Macedonia 41
Statistics Canada 27, 35
Statistics Denmark 5
Statistics Finland 30
Statistics Netherlands 22, 24, 50
Statistics Norway 52

T
Tang, Margaret 27
Taseva, Liljana 41
Thissen, M. Rita 48

U
University of Michigan, The 4, 13, 16, 19, 20, 26, 28, 34
US Census Bureau 36, 37, 53, 54

W
Wensing, Fred 44
Wестат 18, 31