

# Quality assurance through Computer Audio- Recorded Interviewing (CARI): The Statistics New Zealand Case Study

*Chris Seymour, Statistics New Zealand*

## **Background**

In 2006 the Data Collection Standards and Consultancy (DCSC) conducted a research project to identify and establish a set of generic interviewer best practice behaviours for Statistics New Zealand.

The research project recognised that while a number of quality assurance mechanisms were currently utilised within the Statistics New Zealand field collections business unit, they were employed on an ad hoc rather than systematic approach, which was not easily reported on or demonstrated to external parties.

These practices consisted of:

- An annual field check conducted by the interviewer's regional field supervisor.
- Checking of completed interviews on receipt in the office.
- Regular field and completed survey checks of newly recruited interviewers.
- Rotation of interviewers during quarterly or longitudinal surveys so that more than one interviewer collected data from the respondent and anomalies in collection practices could be identified.
- Informal 'teaming' of inexperienced interviewers with more experienced interviewers.
- Training.

The project set about compiling a set of standards brought together from a variety of sources including market research organisations, international statistical agencies and social science research. These standards were then presented to and endorsed by the Statistics New Zealand Standards Governance Board with a view that the implementation of these standards would:

- Increase confidence in data quality, for both stakeholders and the public.
- Increase confidence that official statistics are collected via sound methodology.
- Increase consistency across interviewers, minimising the risk of interviewer related error and falsification of data.
- Increase confidence in the professionalism of our field force.
- Increase respondent cooperation and compliance.

The Best Practice Standards (BPS) for Interviewers initiative, which was part of the wider Best Practice Field Collections project, adopted the DCSC's standards as a foundation from which to develop a revised set of field interviewer best practice standards and practical quality assurance mechanisms, to address the issues raised by the research project.

The BPS initiative identified twelve recommendations to quality assure interviewing standards, of which the use of Computer Audio-Recorded Interviewing (CARI) was one. CARI provides the unobtrusive recording of verbal communication between an interviewer and respondent as the survey is conducted.

## **Computer Audio-Recorded Interviewing (CARI)**

Traditional methods used to ensure the quality of field collected data are often expensive, logistically challenging and difficult to qualify.

Traditional quality assurance method	Problem
Interviewer Rotation	Logistically challenging and expensive for remote areas.
Interviewer Observation	Expensive and time consuming
Post survey respondent phoning	Difficult to qualify results as often the respondent doesn't remember answers

Traditional quality assurance method	Problem
Comparison of survey audit information	Subjective views are reached

The use of a CARI solution aids in meeting the following quality assurance objectives for field collected survey data:

- Quality assurance of the data collected by computer assisted personal interviewing, through
  - Validation of the authenticity of the interview
  - Capture of the full response for a question or set of questions
- Monitoring of adherence to standards by interviewers, through
  - Monitoring of compliance to best practice standards.
  - Providing information to help identify areas for interviewer development.
- Evaluation / confirmation of the effectiveness of questionnaire items in the field
  - Ability of interviewers to read the survey questions in a fluent and understandable manner.
  - Questions elicit the desired response from the respondent (without the need for explanation or probing by the interviewer).

The approach allows for the survey supervisor or administrator to selectively target new questionnaires or questionnaire sections without adversely affecting the information collected as the survey is conducted. The audio information collected through CARI provides evidence that an interview is conducted correctly and to organisational standards, while building up a raft of material to aid in formal interviewer training, which in turn increases the quality of the interviews conducted.

Statistics NZ first became aware of CARI as a quality assurance tool from a demonstration by Rita Thissen of Research Triangle Institute (RTI) at the 2007 International Field Director's and technologies Conference.

Help and guidance from RTI after the conference resulted in the development of a CARI prototype at Statistics New Zealand and has since been further developed into an end to end system.

## Implementing CARI at Statistics New Zealand

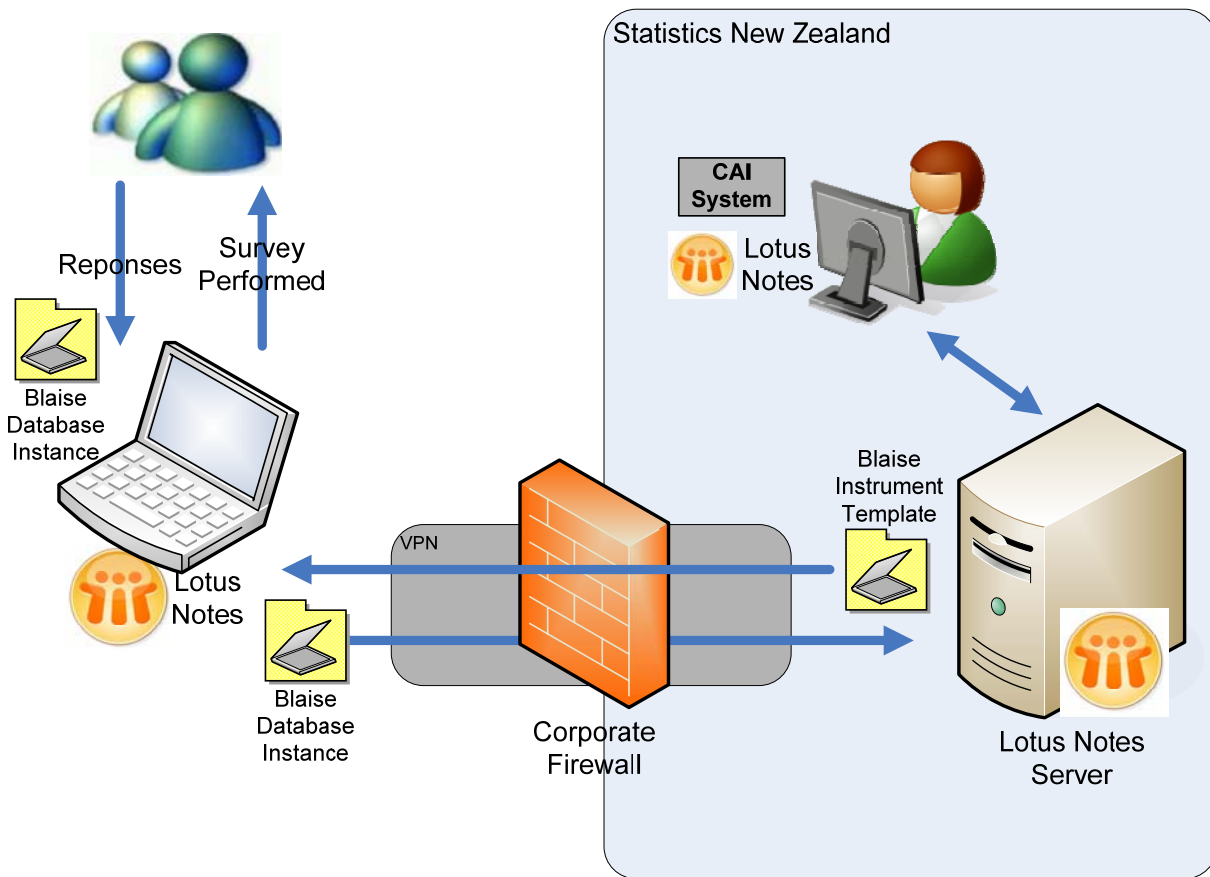
Statistics New Zealand has invested heavily in making Blaise its Computer Assisted Personal Interview (CAPI) tool. This meant that any work done to implement a CARI based solution into our interviewing suite would need to seamlessly integrate with Blaise and other survey management systems.

The current Computer Assisted Interviewing (CAI) system utilises Lotus Notes as a portal that allows collection supervisors to assign cases to the interviewer who will perform the interview.

The interviewers use wireless broadband to download the case list assigned to them and with it the Blaise instrument and associated files to use for that survey.

When the interview begins a instance of the Blaise database is created for the respondent being interviewed and completed as the survey progresses.

Once the interviewer has performed a number of interviews they will replicate with the office system and upload the completed Blaise database into the CIA system. At this time changes to the case list assigned to the collection agent will also be downloaded.



## Solution Design

The first step towards a CARI implementation that integrates into Blaise was to decide the Blaise integration strategy. Two choices are available:

### 1. Component Object Model (COM)

COM is an interface standard created by Microsoft that allows libraries of functionality to be implemented. Using COM allows functionality that is not currently supported in the native Blaise product to be implemented through two extension points available in the Data Entry Program (DEP). These extension points are referred to as:

- Alien Routers – Executed when a question is given the focus within the DEP and that question is tagged with an Alien Router statement.
- Alien Procedures – Executed by the RULES section of the data model within the DEP.

### 2. Extension of the Audit Trail

Blaise provides an audit facility that records the field values and movements within the questionnaire as the survey is conducted.

The source code, written in the development language Delphi, and example audit trail functionality are provided free with the Blaise system.

To help understand these integration options two short implementation prototypes were started, one extending the DEP through the use of the Alien Router keyword, and the second extending the audit trail example provided with the Blaise system.

After careful consideration of the prototype solutions it was decided to continue with the approach that extended the Audit trail functionality. A number of factors lead to this decision:

1. COM based technical issues were experienced resulting in instability of the prototype.
2. Use of the Alien Router approach required that the Blaise instrument be modified to attach the ALIENROUTER keyword to the blocks that will execute the external functionality. The Audit Trail approach required no modification of the instrument.
3. The Audit Trail approach was familiar to our internal Blaise developers as it had been used previously to provide multilingual support for questionnaires.
4. The base audit trail functionality had been stable for a number of years.

### Design elements:

Before any development was done it was identified that the audit trail example, which is the base for our CARI implementation, used a procedural based development style which is arguably more difficult to read and therefore understand, and was a practice that as an organisation were moving away from.

To ensure that this application met our internal development practices and standards it was decided that the code base should be re-written using Object Oriented Programming (OOP) techniques, making the functionality easier to manage, maintain and extend. The remade audit trail code forms the base of the SNZAudit trial, retaining all its original functionality.

One of the known issues in relation to implementing an audio recording feature was the management of the audio files once generated. This posed problems not only for the storage of the audio files on the laptop, but the time and cost required transferring them back into the organisation.

Compressing the audio files was seen as the only option; however this raised a significant technical challenge. After some research a third party product called Active Audio Record (AAC) was found which allowed the audio stream to be recorded using one of a number of formats, including compressed formats like wma and mp3.

In order to identify the audio file generated for a particular question a robust naming convention was created using identifiers from the Blaise instrument and CAI system which together form a unique string that is used as the file name. The format is as follows:

```
<SurveyID><CaseID>__<SurveyType><SurveyVersion>_<CaseID><QuestionnaireID>_<Module>.<Series><index>.<Question><index>.WMA
```

Where:

Format Part	Description	Source
SurveyID	The survey identifier	CAI
CaseID	The unique case identifier	CAI
SurveyType	The survey type, (H)ousehold or (P)ersonal	CAI
SurveyVersion	The survey version	CAI
QuestionnaireID	The Questionnaire identifier	CAI
Module	The name of the blaise module	Blaise
Series	Series is the name of the blaise series or table	Blaise
Question	The name of the blaise question	Blaise
Index	An optional number identifying a repeating or sliding question	Blaise

### Solution Usage

An important aspect of the Blaise integration approach that was adopted is that CARI does not affect the design or development of the actual Blaise instrument. This allows CARI to be used with any new or existing Blaise instrument by simply configuring the Blaise instrument to use our CARI audit trail functionality.

In order to use an Audit Trail routine Blaise must first be configured with the Audit Trail dll to use. To do this the Mode Library Editor is used (from the Control Centre menu -> Tools -> Modelib Editor).



From this dialog ensure the 'Make audit trail' checkbox is selected and specify the SNZAudit.dll as the dll to use.

The SNZAudit trail system has extended the Audit Trail Information file (.AIF) file to provided configuration items at runtime for both the audit and audio recording functionality. Upon execution of the Blaise instrument the system tries to configure itself from a file located in the same directory and with the same name as the blaise data model (.BMI) file. If this file can not be found then a second configuration attempt is made via a file called AuditTrail.AIF. If this file can not be found then the system is configured using default values.

### Solution Configuration

With the CARI functionality built into the audit trial system additional configuration items are available as explained below.

The original options required by the Audit Trail functionality remain unchanged:

Audit Trail	Default	Comment
Enabled	True	Turns Audit trail on / off
Leavefield	True	Records details on leaving a field
Enterfield	True	Records details on entering a field
Action	True	Records details on an action (actions are menu options and control keys)
Keystroke	False	Records every single keystroke
Classify	False	
Lookup	False	
ErrorDlg	False	Records all error's that occur
FilePerForm	False	Creates a audit file for each 'form' (A file gets created called <Primary key> + .adt
Extra	False	Writes 'extra' info like username, metafile, etc.
FlushFile	False	Flushfile means the file is written straight away rather than waiting for the network to write from the buffer...
Mouse	False	Track mouse cursor
PreciseTiming	False	Shows 1000 <sup>th</sup> of a second
KeyTranslation	False	Handles double byte character sets
WriteLean	False	Writes audit information in an abbreviated format
KeyStrokeOneLine	False	Writes keystrokes on only one line in audit file
AuditFolder	Data dir	Folder in which to save audit files
AuditTempFolder	Data dir	Folder to store temporary audit files

Additional configuration items required by Statistics New Zealand's implementation of CARI:

CARI	Default	Comment
Enabled	False	Turns the recording functions on / off
WavAsKey	True	Adds the Key to the filename
Device	First found	Name of the recording Hardware device
Format	First found	MP3, WMA, OGG, VOX, AU, AIFF, MP4, FLAC
WaveFormat	First found	e.g. "5kBPS, 8kHz, Mono"
Mixer	First Found	e.g. "Front Mic"
WriteToAudit	True	Write's into the audit trail recording details.
SeperateFiles	True	Stores data in separate audio files
WavFolder		Folder the audio files are to be written to, if no value is here will default to Audit folder.
StartQs		List of 'start questions' separated by ';' NOTE: This is the FULL question name
EndQs		List of 'end questions' separated by ';'.
MaxTime	60	Max time in seconds for each question.
ScriptStart		The start question of any CARI script
ScriptFile		A 'rich text' file that contains the content text to display.
ScriptConfirmText		The confirmation answer text to display
ScriptCancelText		The cancel recording answer text to display
MaxFiles		Maximum number of recordings per case
Random		Random chance for recording (decimal .5 = 50%)

Optional configuration item used to assist feature development:

Debug	Default	Comment
Enabled	0	Turns on debugging mode. When enabled a dialog window sits over the Blaise instrument displaying the audit trail and other Blaise information in real-time.

### Recording Consent

Before a survey that has CARI enabled is started consent for the interview to be recorded is required from the respondent. To capture consent the interviewer is presented with a Blaise-like screen which presents the content question from the Rich Text Format (.RTF) file configured in the audit information file.

Blaise 4.8 Data Entry - C:\Users\Administrator\Desktop\Test\test

Forms Answer Navigate Options Help Stop Recording Debug On

For quality assurance and training purposes, short parts of this interview will be recorded and listened to by supervisory and quality assurance staff, and questionnaire developers.

Do you give your consent for the recording to occur?

Yes  
 No

CARI 1

Once consent has been granted by the respondent the survey will proceed as normal. If consent is denied or withdrawn at some later point after the survey has begun then a reason for this action is captured through a popup window and stored in a separate file which is associated with this interview.

Recording is manually stopped through an additional menu item attached into the Blaise menu structure.

The action of stopping the recording process causes any audio files recorded up to that point to be removed.

### Recording Questions

A number of options are available to determine exactly how and which questions are recorded when CARI is active.

The list of questions available to be audio recorded is configured through the CARI .AIF file using the 'StartQs' and 'EndQs' configuration items. These items hold a semi-colon separated list of the full Blaise question identifier.

Generally these two lists are the same and cause a recording to be started when the question is entered and stop when the question is left. By specifying different start and end questions within these configuration items enables a range of questions to be recorded. Each start and end set produces a separate audio file.

To provide a random element to which question is recorded a value between 0.05 and 0.50 can be assigned to the 'Random' configuration item. This value determines the likelihood that the question will be recorded. A default value of zero will ensure all the configured questions are recorded.

The length of a given recording can also be determined through the .AIF file. The 'MaxTime' configuration item sets the maximum durations, in seconds, of any individual recording.

The configuration item 'EndRecQ' specifies the Blaise question that informs the CARI system that no more recording is required. When this question is complete a file is created that stores an audio recording complete status. If the interviewer traverses back through the instrument, or the interview is stopped and restarted, audio recording will not be restarted as a check is made to see if the audio recording complete file exists.

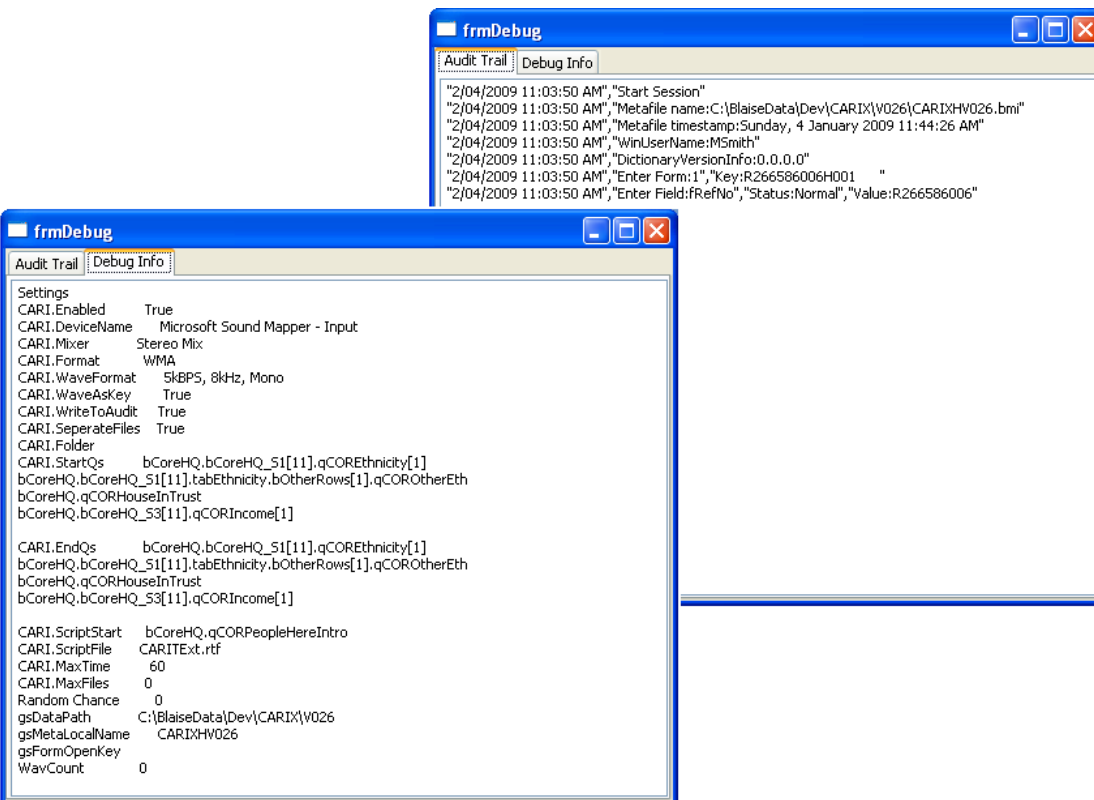
## Recording Format

The recording format used to produce the audio files has a significant impact on the usability of the solution, with factors such as hard drive space and transmission bandwidth being important considerations over the longer term.

Through the configuration item 'Format' one of a number of compressed and uncompressed audio formats can be selected to record the audio stream. WMA was selected as our default audio format as it provided a more efficient compression of voice recording and therefore generated smaller files. However any audio codec that is available to the operating system can be used.

## Solution Debugging

Another useful feature built into the Statistics New Zealand Audit trial functionality is the ability to monitor the audit information in real time. Enabling the configuration item 'Enabled' in the [Debug] section of the .AIF file causes a separate window to be displayed when the Blaise instrument is started.



## Solution Integration

One of the main design goals was to integrate audio recording with Blaise and Statistics New Zealand's field collection applications in a way that caused as little impact and rework as possible to those existing systems.

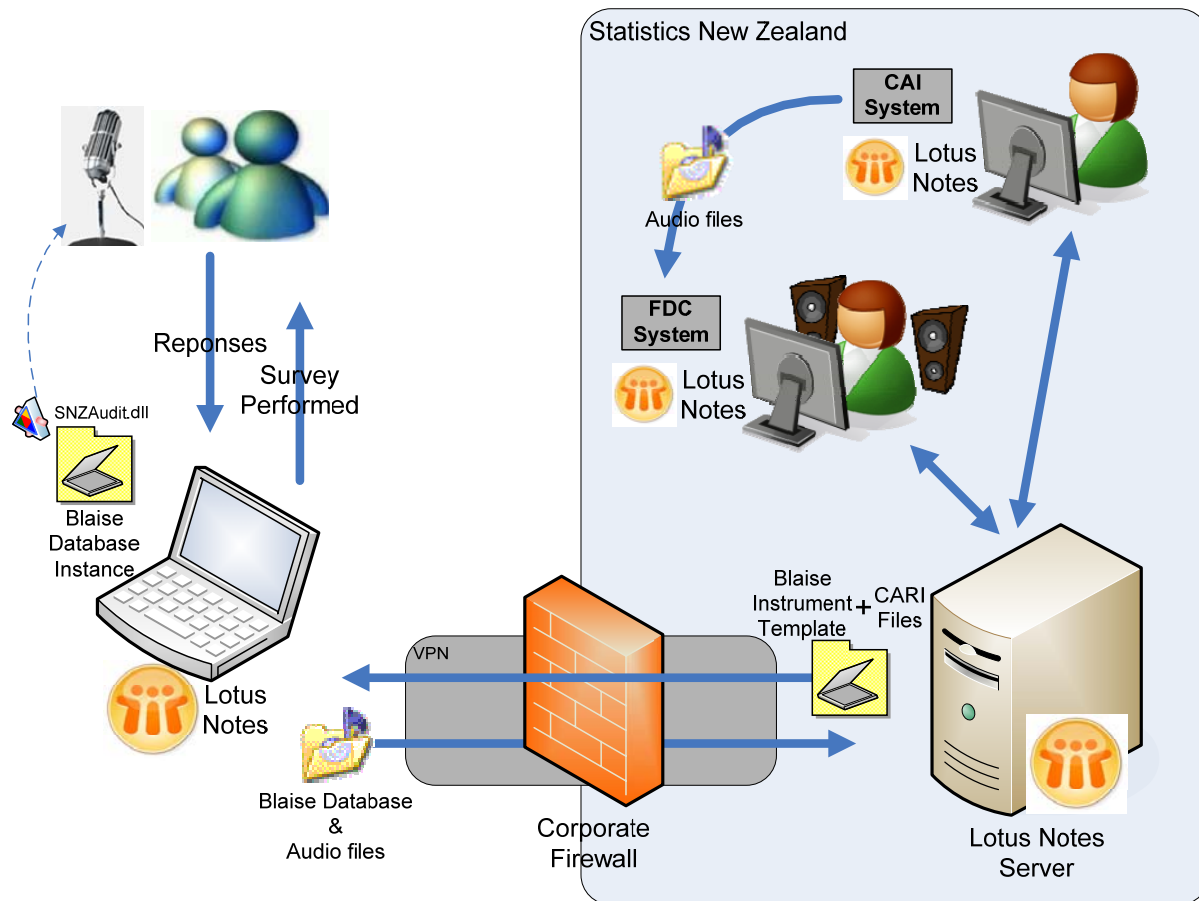
### CARI capture

The CAI system was extended to replicate the CARI configuration file (.AIF file) out to the laptop of the collection agents as the case list and Blaise instrument are transferred.

To enable CARI on a Blaise instrument requires that the instrument is configured to use the SNZAudit.dll as the audit trail library before it is uploaded into the CAI system. This is currently done by the staff responsible for the development of the instrument. Once on the Laptop, security settings prevent respondent from changing any options within the Blaise instrument or CARI. Any CARI related files or audio recordings can not be seen or accessed by the interviewer.

The CAI system was further extended to replicate the audio files created through the interview process back into the organisation along with the completed Blaise instrument and attached back into the CIA system.





### CARI Management

Once the audio recordings are received within the organisation, quality assurance staff use the audio recording to rate the interview and interviewer performance in association with other field collection best practices

To assist the review process another application called the Field Collections Directory (FCD) was extended to provide additional functionality that enables quality assurance staff to listen to the audio recording made by CARI and provide a rating or comment on the effectiveness of the interviewer in asking this question.

The FCD also enables questionnaire developers to use CARI recordings to evaluate question performance.

An example of the CARI Case Review screen is shown here. Note the review questions shown below are sample only and can be changed by the Field Collections Manager. The overall question rating is determined from the answers selected by the reviewer.

Interviewer	CAI Demo2	Comments	Overall very well done
Region	Region 0		
Date	26/03/2009 16		
Survey	CARI		
Case ID	R469553004		
Type	<input checked="" type="radio"/> Routine <input type="radio"/> Targeted		
CARI Case Status	Active - Incomplete	Follow-Up Required	<input type="checkbox"/>
Overall Case Rating	80.00%		

1 of 3

Question ID	CARIXR469553004__H027_R469553004H001_bCoreHQ.bCoreHQ_S1[11].qCOREthnicity[1].WMA		
Question text	Which ethnic group or groups do you belong to?		
Answer recorded	Cook Island Maori		
CARI Review Questions	Does the interviewer read the question exactly as worded in entirety, including all alternatives?	<input checked="" type="radio"/> Yes <input type="radio"/> No <input type="radio"/> N/A	Comment
	Does the interviewer record the answer as given?	<input type="radio"/> Yes <input checked="" type="radio"/> No <input type="radio"/> N/A	Comment
	Are the probing guidelines followed?	<input checked="" type="radio"/> Yes <input type="radio"/> No <input type="radio"/> N/A	Comment
	Does the interviewer's response to queries bias the respondents response?	<input type="radio"/> Yes <input type="radio"/> No <input checked="" type="radio"/> N/A	Comment
	Does the interviewer have a slow and distinct speech which allows the respondent to clearly understand the survey questions?	<input type="radio"/> Yes <input type="radio"/> No <input checked="" type="radio"/> N/A	Comment
Question Rating	66.67%		

## Evaluating the CARI implementation

### Progress

While CARI appears to be a relatively simple concept, the amount of work required to develop and integrate an end-to-end solution should not be underestimated.

In addition to the development of the technical solution a number of other considerations have come to the surface, including:

- Interviewer concerns related to a potential impact on response rates.
- File sizes and the impact on replication – replication takes longer, therefore cost goes up.
- Privacy issues relating to recording respondent responses – ensuring that the audio recordings are secured, both on the interview laptop and within the organisation.
- Procedures to govern and manage the audio recordings over time – Archiving of the audio recording.

While some of these issues have been addressed others have yet to have their impact fully determined as the solution, although complete, has not been tested in the field. Statistics NZ are currently working on identifying an appropriate survey to begin a trial.

It is seen that CARI will meet several critical needs both those specific to field interviewing and in general, CARI provides a means to remotely monitor the quality of the field interview, including adherence to standards during the interview and the reactions of the respondent to survey questions. Statistics New Zealand expects to utilise CARI to assist:

- Evaluating interviewer performance and providing feedback to interviewers
- Identifying questionnaire problems and opportunities for survey improvement
- Detecting interview fabrication and interview errors

### Lessons

A change to the operating system on the interviewer laptop's, from Windows XP to Windows Vista, while the CARI solution was being system tested caused a number of issues and ultimately a delay to the intended field test. It is recognised that both the development, system test and production environments must be comprised of the same software to avoid these kinds of problems in the future.

Looking forward it is recognised that there is limited internal knowledge of the development language (Delphi) used to create the CARI application. This poses risk for the on-going support and maintenance of the solution

and it is therefore envisaged that the solution will be redeveloped using the current corporately supported development toolset of Microsoft's .Net framework.

Overall CARI aligns with the Statistics New Zealand strategic goals of statistical excellence, integrity and leadership;

- Statistical excellence - CARI will help ensure that the Field Collections business unit produces relevant and accurate data through improved technical and quality standards.
- Integrity - CARI will improve the transparency of field-based data collection and should lift the level of trust subject matter clients and data users have in the resulting outputs.
- Leadership - CARI has become an accepted quality assurance mechanism internationally and Statistics New Zealand will be leading its adoption in New Zealand.

It is envisioned that CARI will form a key part of the Field Collections' quality assurance programme and is central to the systematic monitoring of data collection in the field and will provide robust, timely information to inform Field Collection decisions and processes now and into the future.

## References

Wikipedia, Component Object Model (COM) definition, 2009,

[http://en.wikipedia.org/wiki/Component\\_Object\\_Model](http://en.wikipedia.org/wiki/Component_Object_Model) accessed 4/4/2009

Mitchell, CARI Audit Trail Information Sheet, 2008, *Statistics New Zealand internal document*

Hill Christine, Best Practice Field Collections - CARI Software Specification, 2008, *Statistics New Zealand internal document*

Broadhurst Donna, Best Practice Standards (BPS) for Interviewers - CARI Functionality and Requirements, 2008, *Statistics New Zealand internal document*

Broadhurst Donna, Best Practice Standards (BPS) for Interviewers - Quality Assurance Vision Document, 2007, *Statistics New Zealand internal document*

*Methods and Informatics Department* Statistics Netherlands, Blaise 4.5 Developers Guide, 2002

# Development of Survey and Case Management facilities for organisations with minimal survey infrastructure

*Fred Wensing, Softscape Solutions, Australia*

## 1. Introduction

In order to conduct a survey you need more than just a questionnaire and a computer to run it on. You also need to have a system which can manage your survey and manage the case records that will be created.

Organisations that run surveys regularly will probably have existing infrastructure which can be used to manage the conduct and operation of those surveys. For them it will be sufficient to modify their interfaces to interact with Blaise questionnaires. For smaller organisations, or ones with limited survey infrastructure, however, it will be necessary to develop some facilities that can provide some survey and case management functions.

This paper describes a basic survey management system that has been developed for use by three organisations that had minimal or no survey infrastructure before choosing to use Blaise for their surveys.

## 2. Typical situation

The three organisations involved were:

- The Australian Department of Family, Community Services and Indigenous Affairs (Longitudinal Study of Indigenous Children)
- Telethon Institute for Child Health Research
- New Zealand Ministry of Tourism

The typical situation that I found with these three organisations was:

- They want to conduct a survey relevant to their needs
- They want to use computer assisted interviewing
- They have heard of Blaise and obtained a licence (or evaluation licence)
- They undertake to develop or commission a Blaise questionnaire

When they get further down the track, they plan to:

- Purchase notebook computers for the survey
- Employ interviewing staff to conduct the survey

It is not until the field operations were considered in detail, however, that it was realised that some additional infrastructure would be necessary to help manage the survey. In general, these organisations had little or no existing survey infrastructure to assist with the conduct of the survey. The common assumption was that Blaise software could provide these facilities.

There is no ready-made case management facility provided with Blaise software, but the tools for developing such facilities do exist.

## 3. Choice of tools to develop management facilities

It is possible to develop survey and case management facilities either by using the Maniplus language of the Blaise suite or by using the Blaise API component. In Blaise 4.81 there is also the Blaise Data Centre utility which may have a role.

Maniplus has the advantage that it is part of the Blaise suite and can readily access Blaise data files. The Maniplus language provides relatively easy ways to program dialog elements containing fields and buttons that can be activated or hidden, depending on the situation. The development can be done using the Blaise Control Centre and the skill level is marginally more than needed to develop Blaise questionnaires.

To make use of the Blaise API component it is necessary to develop a separate application using a scripted language (eg. Microsoft Visual Basic, C++ or .NET) that can provide you with an interface which then can be used to interact with Blaise data files. This requires additional development software (eg. Microsoft Visual Studio) as well as suitably skilled programmers to carry out the development. There are also additional licence fees to be able to use the Blaise API component.

The Blaise Data Centre provides an interface to examine Blaise data and carry out some case management functions such as deployment and importation of cases. The facility is interactive and does not seem to provide a programmable interface so that you can tailor it to particular needs. There are also additional licence fees to be able to use the Blaise Data Centre. This utility would have functional use in the office environment and could complement facilities set up on the notebook computer.

It was decided that Maniplus would be used to develop survey management facilities because none of the organisations had ready access other software or skilled programmers. It would also be easier for survey staff, with some skills in Blaise, to maintain a Maniplus application once it was developed. It was also part of the basic Blaise software installation.

## 4. Facilities needed

When considering the facilities that needed to be developed for the notebook computer it was decided to focus on features that were essential for the direct operation of the survey. General business issues such as secure access to the computer, internet access and communication with the office were considered out of scope because they could be handled by existing software and protocols.

The following features were considered important and necessary to be incorporated into a basic survey and case management system that could operate on the notebook computer:

- Establishment of the survey environment
- Identification of the operator
- Installation and removal of surveys
- Display of current surveys
- Display of cases for the selected survey
- Management of the interview entry and exit
- Managing the status of cases
- Packing up of cases for return to the office

## 5. System design

The system was designed to have three layers of management programs.

The top layer is the Survey Manager program which provides the single point of entry for the operator to access all surveys on the notebook computer. From within the Survey Manager the operator can install or remove surveys, and/or select one to be used for the current session.

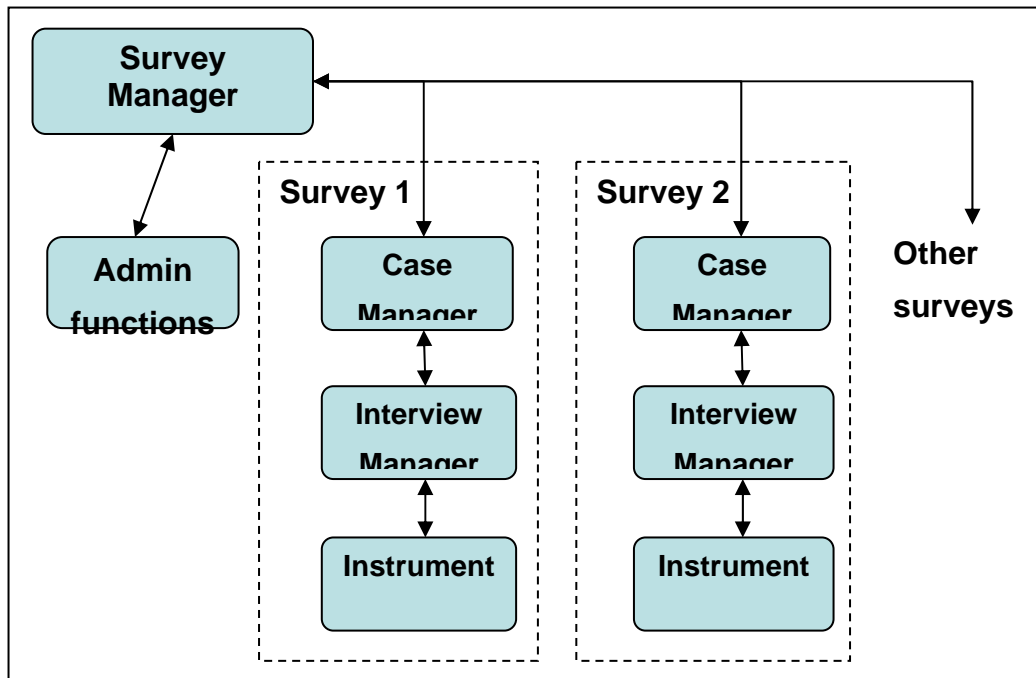
The second layer handles the Case Management functions relevant to each survey. A survey would have its own Maniplus program tailored to its particular needs.

The third layer is the Interview Management layer. Once again, each survey would have its own Maniplus program that handles the entry into and exit from the interview for a selected case record.

Underneath the management layers are the Blaise instruments (electronic questionnaires) that are used in the surveys.

The high level system design is illustrated in Figure 1. Each facility is described in more detail in the sections that follow.

Figure 1. High level system diagram showing three layers of management



## 6. Survey environment on the notebook computer

The main requirement for the survey environment on the notebook computer is to keep it as simple as possible. That way, it should be relatively easy to resolve any problems that might arise.

### 6.1 Blaise Software installation

It was decided that the easiest way to set up Blaise on the notebook computers to be used by the interviewers was to install a current complete version of Blaise using the standard Blaise installation package. That way all the necessary software elements, and more, were available on the computer should they be needed.

While it is possible to operate Blaise surveys using a cut-down version of the software (involving a small number of EXE files from the standard installation) that would have required a special installation package to be built. There were no resources to develop or maintain such a special installation package so the full installation was chosen.

It is not necessary to include the Blaise licence key in the installation because the interviewers do not need any of the licensed components.

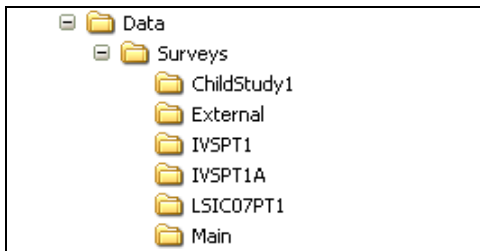
In the event that survey managers are concerned about interviewers running parts of the Blaise software which they should not, it is possible to remove access to the Blaise Control Centre by removing the Blaise.EXE file from the installation. This would have to be done manually.

### 6.2 Folder structure

In order to manage the surveys to be conducted on the notebook, a simple folder structure was devised.

The primary folder under which the survey manager application and all surveys would be placed was called 'Surveys' and this was placed in a folder on the C drive called 'Data'. The Survey Manager application was placed in a folder called 'Main' and this was located in the 'Surveys' folder. Each survey would then be placed in its own named folder, also in the 'Surveys' folder. In order to share the use of external look-up files across multiple surveys, an 'External' folder may also be useful. See Figure 2 for a screen-capture of the folder structure.

Figure 2. Screen-capture view of survey folders once 4 surveys have been installed



There is no need to set up these folders because the installation processes take care of that.

## 6.3 Other software

No other software is required to conduct Blaise surveys but it would be expected that some standard software would be available to handle security, internet connection and communications (eg. E-mail client software).

## 7. Survey Manager application

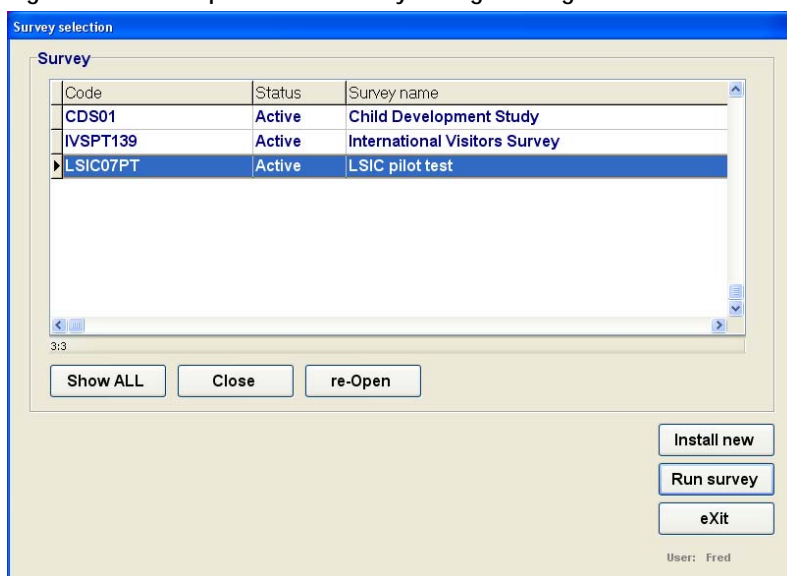
The Survey Manager application provides:

- Single start-up point
- User identification
- Survey/instrument installation
- List of available surveys (for selection)
- Buttons to start a session using a selected survey

### 7.1 Interface

The interface consists of a screen that lists all available surveys with buttons for installing new surveys and activating current ones (See Figure 3).

Figure 3. Screen-capture of the Survey Manager dialog



## 7.2 User identification

User identification occurs when the Survey Manager application is run for the first time. At that point a small dialog screen is provided showing a field into which the user can enter his/her user name.

Once a user's identity has been recorded then that is passed down to the other management programs and can eventually be inserted into the case record to identify the operator who has been collecting or working with the data.

It is also possible to assign different roles to different users by storing a list of available users for a survey.

The user identification developed for this application is fairly open. Once a name has been given it is stored and all future uses of the application on that computer assume the same user name. Essentially the application is identifying the computer rather than the operator but this was considered adequate for the needs of the organisations involved.

Access to the Survey Manager application is not restricted in any way as that was considered unnecessary because access to the notebook computer was presumed to be controlled by normal Windows user and password control.

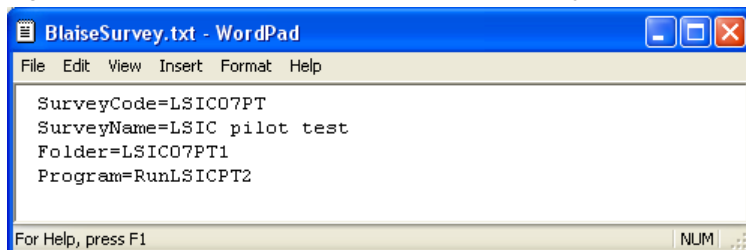
## 7.3 Installation of a survey through the Survey Manager

The interface provides a button that enables the operator to install a new survey into the system. A new survey would generally be supplied on CD or USB drive which would contain a folder holding all required Blaise instruments, Maniplus programs and any other files needed for the survey.

To control the installation process a small text file called **BlaiseSurvey.TXT** is included with the installation files (see Figure 4). This text file contains the following four items of information

- The full name of the survey
- An identifying code name for the survey
- The target folder name
- The name of the Maniplus case management program to be run

Figure 4. Screen-capture of the content of BlaiseSurvey.TXT



The installation process uses the supplied information to create the target folder and copy the survey files into it. Once the survey files are installed the list of available surveys is updated to show the new survey.

The Survey Manager is set up to prevent the same survey (ie. one with the same Survey identification code) from being installed a second time. This was done to avoid accidentally over-writing collected data.

## 7.4 List of surveys

The Survey Manager maintains two lists of surveys for the operator. One list contains the names of all surveys ever installed and the other contains the list of active surveys only. The default list is the list of active surveys. Surveys can be moved between the lists using the **Close** and **Re-Open** buttons.

Surveys are never deleted by the system. If a survey actually needs to be deleted then that could be done manually next time the notebook computer is in the office.



## 7.5 Running a survey from the list

Once a survey is selected then it can be run from the Survey Manager using the **Run Survey** button. When that is done then the system executes a command to the corresponding Maniplus Case Management program, passing in the user name as a parameter. Control is then passed to the Case Management program. When that program finishes then control returns to the Survey Manager.

## 7.6 Installation of the Survey Manager

The Survey Manager is installed from a CD or USB drive which contains the program and simple Install.BAT file. The install process creates the target folder and copies the program elements.

# 8. Case Manager applications

The Case Manager application provides:

- Access to a List of cases for the survey
- Ability to operate on the list (load, remove, package)
- Buttons to start an interview using a selected case
- Management of the case records in the survey folder

Each survey generally has a different set of needs and each will therefore require a tailored program. The sections that follow discuss some of the issues and how they can affect the Case Management program.

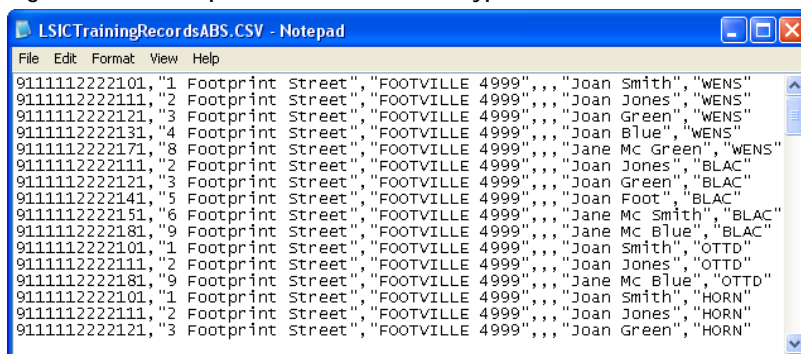
## 8.1 Pre-defined and allocated list

Many surveys operate on a list of addresses (or cases) which are allocated to particular interviewers.

To manage this kind of survey it is important to maintain a single central list (in the office) which keeps track of the case identifiers and which interviewer has been assigned each case. Whenever a case is moved from one interviewer to another the list needs to be updated in the central repository.

From time to time the updated case list, in the form of a comma-separated text file, would then be sent to the interviewers for loading. Figure 5 shows an example of such a list.

Figure 5. Screen-capture of the content of a typical case list

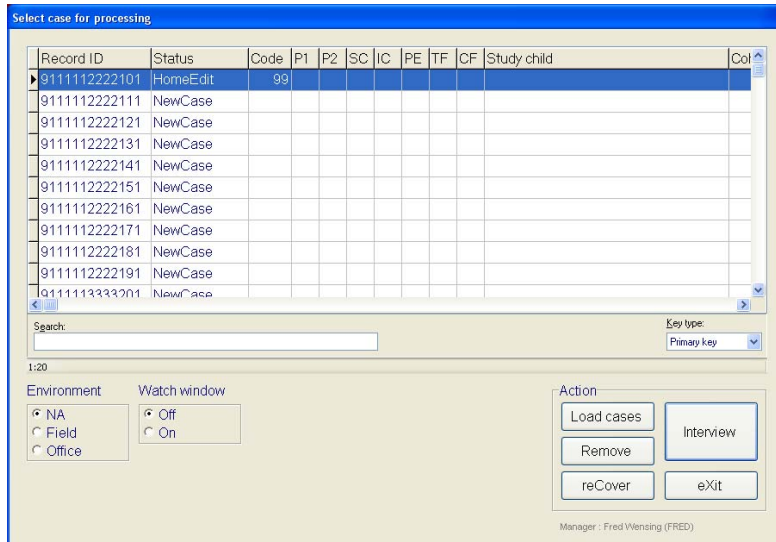


The Case Manager application for such a survey should be able to load the latest case list whenever it is updated. The system also needs to keep track of all the case lists that have been loaded so that it does not re-load an old list.

Where cases are allocated to specific interviewers, the Case Manager would need to select, for display, only those cases that match the user name.

Figure 6 shows a screen-capture of the Case Manager developed for the Survey of Indigenous Children where cases were assigned to particular interviewers. Note the button for loading updated case lists.

Figure 6. Screen-capture of the Case Manager where an assigned list is involved



For convenience of the operator, options exist in the Case Manager interface to remove completed cases from view.

## 8.2 Dynamic sample

For some surveys, there is no predefined sample. Instead, the sample is defined at the time of interview. This can happen when there is random selection at a venue such as an airport. In this situation the Case Manager needs to create uniquely identified cases.

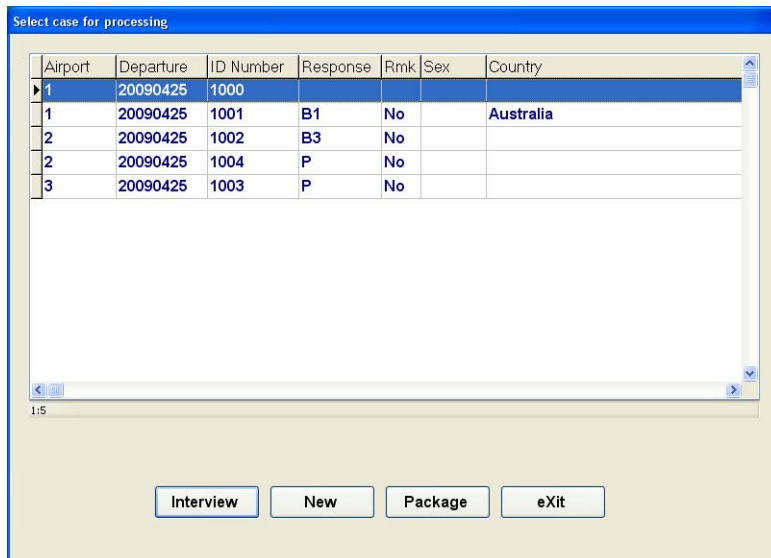
The International Visitor Survey run by the New Zealand Ministry of Tourism is just such a survey. The Case Manager was programmed to set up unique identifiers based on airport, current date and interviewer code. At the commencement of each session the operator is required to identify the airport and date (see Figure 7). The system keeps track of the last used case number and assigns new numbers to new cases.

Figure 7. Screen-capture of the session identification dialog



New cases are added to the case list as the interviewing session progresses. Figure 8 shows a screen-capture of the Case Manager developed for the International Visitor Survey where cases were created dynamically using the **New** button. Previously created cases can be accessed using the **Interview** button.

Figure 8. Screen-capture of the Case Manager for a dynamic sample



### 8.3 Management of case records in the survey folder

There are two ways to handle the Blaise case records which are created:

- Have a single file containing all case records, or
- Treat each case record as a file in its own right

The single file approach is satisfactory provided that there is no need to move individual cases between interviewers. This was the case with the International Visitor Survey.

However, as soon as there is a possibility that cases may need to be moved around then it is best that each case record be stored in a separate file. This was the situation with the Longitudinal Study of Indigenous Children.

#### 8.3.1 Managing a single file of all records

Using a single Blaise data file to hold all the case records is relatively easy provided that each case has a unique identifier. The Blaise file is set up with the corresponding primary keys for the unique identifier and access to the record is done by using the relevant key.

No special file management is required until the data needs to be sent back to the office. This is discussed in section 8.5.

#### 8.3.2 Managing separate files for each case record

Using separate files for each case record does make the Case Manager programs more complicated to write, but gives the advantage of being able to manage each case separately.

Management of the cases is handled by using the ZIP functions of Blaise (available from version 4.8 onwards) or by using the Winzip command line utilities.

Essentially, once an interview is closed the Case Manager collects all the relevant Blaise files and copies them to a Zip file for storage. If the case record is revisited then the Case Manager removes the Blaise files from the Zip file and places them back into the system before running the Interview Manager program.

The relevant Blaise files are identified using the Blaise database name with the \* wildcard to pick up all files. Metadata files are excluded from the Zipping process through the use of a list of exclusions, recorded in a text file called **Exclude.TXT**.

Figure 9 shows the procedure used to Zip the data for one case record (using Blaise commands). In this procedure **IDCode** is the identifier of a case and **aFormName** is the name of the survey instrument (Blaise

datamodel) and **Exclude.TXT** contains a list of files to be excluded from the Zip process. Note that the Audit trail file is also added when present.

Figure 9. Procedure to Zip current data

```
{Procedure to Zip current data file to backup folder using Blaise commands}
PROCEDURE Run_Zip
  DAYFILE('Zip current files to '+IDcode+'.Zip')
  aResult := ZIPFILES ('Backup\'+IDcode+'.zip '+aFormName+'.* /X@Exclude.txt')
  IF ADTEXists=1 THEN
    DAYFILE('Zip ADT file to '+IDcode+'.Zip')
    aResult := ZIPFILES ('Backup\'+IDcode+'.zip '+IDcode+'.adt')
  ENDIF
ENDPROCEDURE
```

Before an interview is started, any existing survey data file is copied to another Zip file called Remove.ZIP then deleted. This is important to ensure that new records are not adversely affected by old data.

Note that unlike the Winzip command line program the ZIPFILES command does not have a 'Move' option that enables removal of the data files but not the datamodel. For that reason the program then deletes all the files then restores the datamodel from the Remove.ZIP file.

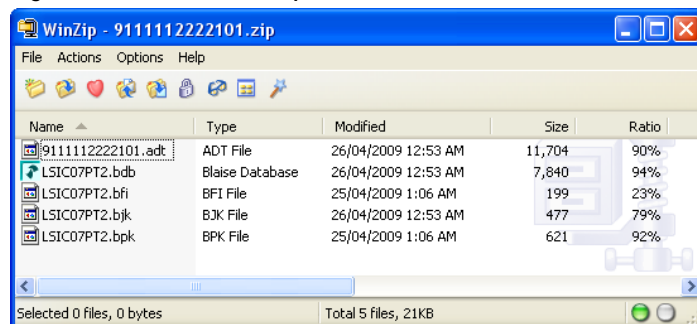
Figure 10 shows the commands that are used to remove any current data and restore a selected case from a Zip file (when it exists). In this part of the program **IDCode** is the identifier of a case, **aFormName** is the name of the survey instrument (Blaise datamodel), **aInstallFolder** is the location of the current survey installation and **Exclude.TXT** contains a list of files to be excluded from the backup Zip process.

Figure 10. Segment of source code showing extraction of case data from backup files

```
{First remove any (old) data files that may be present}
IF FILEEXISTS(aFormName+'.bfi') THEN
  DAYFILE('Clean up files to Remove.zip')
  {Copy the instrument files to a local Zip}
  aResult := ZIPFILES ('Remove.zip '+aFormName+'.*')
  {Delete all survey files}
  aResult := RUN ('DEL '+aFormName+'.*')
  {Extract the survey datamodel from the Remove.Zip file}
  aResult := UNZIPFILES ('Remove.zip /D'+aInstallFolder+' @Exclude.txt')
ENDIF
{If a backup file exists then extract the data and use it}
IF FILEEXISTS('Backup\'+IDcode+'.Zip') THEN
  DAYFILE('Unzip files from '+IDcode+'.Zip')
  aResult := UNZIPFILES ('Backup\'+IDcode+'.Zip /D'+aInstallFolder+' *')
ELSE {otherwise create a new record}
```

Figure 11 shows the content of a single Zip file containing the Blaise data files for one case record and the corresponding Audit trail file. Notice that only the data files are present and that the datamodel is not included.

Figure 11. Content of the Zip file for one case record



## 8.4 Running the interview(s)

Once a case record has been selected for interview and the data restored (if present) then the Case Manager hands control over to the Interview Manager program, passing in parameters to identify the user and the case.

## 8.5 Packaging the cases for sending to the office

Depending on the survey, the Case Manager can also be programmed to carry out other case administration tasks such as packaging for return to the office.

The packaging process is a separate Maniplus program that makes use of the ZIP functions of Blaise to put all cases for a designated survey into one Zip file. This makes it easier for the operator to send the data back because the program has made the selection of files and it produces just one file to be attached to an e-mail.

The packaging process is activated through the **Package** button on the Case Manager screen.

Because the packaging process removes those records from the system, the operator is warned that they will no longer have access before execution takes place.

When the packaging is complete a small report is shown on the screen for the operator (see Figure 12).

Figure 12. Screen-capture of the packaging report



## 9. Interview Manager applications

The Interview Manager application provides:

- Program control over the entry and exit from the interview
- Check of the clean/dirty status of the record and inserts the status value into the data
- Handling of the setting of response status

### 9.1 Controlling the entry and exit from the interview

The Interview Manager is a Maniplus program that provides a shell around the interview. The program accepts parameters from the Case Manager and then starts up the interview for the identified case using the Manipula EDIT command.

The following command line options are applied to the EDIT command to:

- Identify the case to be opened (option /K)
- Specify the Menu file to be used (option /M)

- Identify whether the Audit trail is to be activated (option /C)
- Turn on the watch window if required (option /!)
- Exit the data entry process when the interview is closed (option /X)

Other options may be added if needed but these are the common ones.

Figure 13 shows a typical activation of the interview using the EDIT command. In this part of the program **aWatchW** is for the watch window setting and **aRecordID** contains the identifier of the case record as supplied via parameters from the Case Manager.

Figure 13. Segment of source code showing the start-up of the interview

```

{Set the watch window based on Parameter 2}
IF PARAMETER(2) = 'ON' THEN
  aWatchW := '/!'
ELSE
  aWatchW := ''
ENDIF

{start-up the interview}
IF PARAMETER(3) <> '' THEN
  {apply the Audit trail settings}
  aResult := fInterview.EDIT('/MIVS.bmf /X '+aWatchW+' /K'+aRecordID
    +' /C'+PARAMETER(3)+' .DIW')
ELSE
  {without the Audit trail}
  aResult := fInterview.EDIT('/MIVS.bmf /X '+aWatchW+' /K'+aRecordID)
ENDIF

```

The most important function that the program provides, however, is to take over when an interview is closed. This is covered in the next two sections.

## 9.2 Checking the clean/dirty status

Once the interview is closed it is possible for the Interview Manager program to check whether the record is clean or dirty. This is done by examining the FORMSTATUS of the record (see Figure 14). It is then possible to record that status in a field in the record for ease of access by users or other systems.

Figure 14. Segment of source code that checks the clean/dirty status

```

{check the instrument status and record the result in the BlaiseStatus field}
IF fInterview.FORMSTATUS = NOTCHECKED THEN
  fInterview.CHECKRULES
ENDIF
IF fInterview.FORMSTATUS = CLEAN THEN
  fInterview.BlaiseStatus:=BLClean
  aStatus := 'CLEAN' {Instrument status is CLEAN}
ELSEIF fInterview.FORMSTATUS = DIRTY THEN
  fInterview.BlaiseStatus:=BLDirty
  aStatus := 'DIRTY' {Hard errors exist in the instrument}
ELSEIF fInterview.FORMSTATUS = SUSPECT THEN
  fInterview.BlaiseStatus:=BLSuspect
  aStatus := 'SUSPECT' {Soft errors exist in the instrument}
ELSEIF fInterview.FORMSTATUS = NOTCHECKED THEN
  fInterview.BlaiseStatus:=BLNotchecked
  aStatus := 'NOT CHECKED' {Instrument status is NOT CHECKED}
ENDIF

```

### 9.3 Setting response status

Once the interview is closed the Interview Manager can intervene and provide the operator with a screen where the response status can be set.

The response status options are modified to match the completeness of the interview. The logic for completeness is best done within the Blaise instrument because then it is available to any system that needs this information. Completeness can be tested by placing particular points within the instrument flow then checking how many of those points have been triggered (see Figure 16).

Depending on the completeness of the interview, then only particular response status options are possible. The response status screen is set up so that only the relevant response settings are activated and others are not (see Figure 17). At this point there is also an option for the operator to re-enter the case record if desired. Remarks can also be added in the Response Status dialog and these will be inserted into the case record.

Figure 16. Segment of source code with typical logic for completeness of an interview

```
IF StartInterview=Continue THEN
  IF Endsurvey=Continue THEN
    IF Eligible=No THEN
      StatusFlds.Condition1:=3 (ineligible)
    ELSE
      StatusFlds.Condition1:=1 (full response)
    ENDIF
  ELSE
    StatusFlds.Condition1:=2 (partial response)
  ENDIF
ELSE
  StatusFlds.Condition1:=4 (empty record)
ENDIF
```

Figure 17. Screen capture of the response status dialog with relevant entries activated

Select Response Status

**Codes available**

- A. Complete interview
- B1. Partial interview, refused**
- B2. Partial interview, boarding**
- B3. Partial interview, other**
- C. Eligible, refused
- D. Eligible, refused entry
- E. Eligible, boarding
- F. Eligible, language
- G. Eligible, other
- H. Unknown, refused
- I. Unknown, boarding
- J. Unknown, language
- K. Unknown, other
- L. Missed, no interviewer available
- M. Missed, other
- N. Ineligible (Out of scope)
- O. Quota full
- P. Other

**Response remarks:**

Initial response status: **B1**

Re-enter    eXit

**Information**  
No remarks in this interview.

Upon completion of the Response Status dialog, control is returned to the Case Manager.

## 10. Summary

The programs described in this paper illustrate the kind of facilities that are needed to manage one or more surveys that involve the use of notebook computers in the field. It shows how there are three layers of management that need to be supported: survey management, case management and interview management.

For organisations that have little or no existing survey infrastructure, Maniplus clearly has sufficient functionality and versatility to provide the solution.

