USING BLAISE FOR A CUSTOMER SATISFACTION SYSTEM

Ken Brewster, SIA Ltd, London, UK.

Abstract

The basis of this paper is to demonstrate the flexibility of the Blaise System and how well it links to other commonly used software. The application is a Customer Satisfaction System in which customers are interviewed upon leaving a shop. They are asked a series of questions relating to the products purchased, quality of service, etc. The main aspects of the system are:

(i) a front-end interviewer menu
(ii) a Blaise interview
(iii) quota sampling
(iv) external databases
(v) data transfer for analysis
(vi) letter generation

The technical aspects of the system are discussed, and so a working knowledge of Blaise is assumed. The emphasis shall not be on the interview itself, but on the other Blaise tools utilised, ie. Manipula, Conversions and the Setup Generator.

1. Introduction

The starting point in the system is a Blaise program, ie. the customers are interviewed upon leaving a shop and their responses are stored using Blaise. The population of respondents is constrained by the following:

SEX	AGE	NO. REQUIRED
Male	Under 50	10
Male	50+	5
Female	Under 50	30
Female	50+	15

Once the data has been collected, some standard analyses are to be performed and where applicable, follow-up letters are to be sent out.

2. Technical aspects of the Customer Satisfaction system

The following sections explain how each aspect is handled.

2.1 Front end interviewer menu

The Blaise interview program is essentially an executable file (.exe) and a few associated reference files. This therefore enables Blaise programs to be run from the DOS prompt as they do not require the presence of the main Blaise system. Given this situation, it is therefore possible to build command or batch files to activate the program with a single keystroke or by selecting an option on a menu.

In the case of this system, a menu has been used as shown below.

Figure 1 : The interviewer menu

```
ENTER DATA
EDIT DATA
ANALYSIS
LETTER GENERATION
```

This menu has been developed using Clipper, although any DOS based product could have been used. The main advantage of using menus of this nature is that they can, if well constructed, prevent interviewers from "playing around" on the computer. The Blaise program here is activated by selecting the "ENTER DATA" option on the menu using the cursor keys and then pressing the <ENTER> key.

2.2 Quota sampling

In the market research world, quota sampling is a commonly used technique. A simple example has been incorporated here where the client only wants to collect interviews from the proportions of people as outlined in the introduction.

The program has to be structured in such a way that once the desired level (quota) has been reached in a particular group, no further interviews for that group can take place. Unfortunately, unlike some other CAI products, Blaise does not (yet!) have a dedicated quota sampling facility. The desired result can be achieved though with a few lines of code and the use of Manipula.

To achieve our aim, we need to know, or have access to, the current numbers for each quota group at the start of each interview. This can be done by running a Manipula setup sketch such as that shown below to read the Blaise datafiles, calculate the figures and store them in an ASCII file on disk. Blaise again makes this easy by writing a default Manipula sketch for you, using the Setup Generator. This means that all the user has to do is edit the file to suit the current need rather than write one from scratch. The resultant file will contain a mere four lines (one for each quota group) with three entries on each; the sex variable, the age variable and the current number of entries in the datafile. This file can then be read at any stage during the interview using the READLIST command.

Figure 2a. The Manipula setup file

```
INPUTFILE "CS" BLAISE
SexQ 20     1 INTEGER 0
AgeQ 22     1 INTEGER 0
```

```
OUTPUTFILE "CS.ASC"
     SexQ   1       1;
     AgeQ   2       1;
     Count  3       2 INTEGER;

MANIPULATE
     COMPUTE Count:=1;
     SELECT;

SORT
     SexQ, ASCENDING;
     AgeQ, ASCENDING;
     Count, SUM;
```

Figure 2b. Example of the resultant file

```
11  3
12  5
21  1
22  1
```

Once this information has become available, the questions in the program can be routed accordingly, ie. skipping the main body of the interview if the target value for the current group has been achieved.

Manipula can also be executed from the command line and so to further facilitate smooth running for the interviewer, the command can be placed in a batch file with the interview program command as shown below.

Figure 2c. The batch file

```
@echo off
customer cs /1
manipula cs.man
menu
```

This will start an interview (the key value being automatically assigned) and upon conclusion run the Manipula setup to recalculate the new quota levels, before returning to the main menu. On a computer with a reasonable processor (386 or above), this process is so quick that interruption to the interviewing is negligible.

2.3 External databases

The speed with which Blaise can access information held in external files is very impressive. In this example approximately 5000 address records are held in a separate file. This file is to be cross-checked during the interview to determine whether the respondent had been interviewed previously and to notify the interviewer of any such occurence.

The address file itself starts as an ASCII file but is then converted into an indexed file using the EXTERNAL FILES utility within Blaise. This new file is defined within the program with an external paragraph such as that shown below:

Figure 3. The external paragraph

```
EXTERNAL
   TYPE
          AddRec = RECORD
          Address  = STRING[15];
          Dummy[24];
          PostCode = STRING[11];
   ENDRECORD;
VAR
     A2Idx  :     INDEXFILE "Add2.Idx" [11];
     A2Dat  :     DATAFILE"Add2.Dat" OFAddRec;
     A1Idx  :     INDEXFILE "Add1.Idx" [15];
     A1Dat  :     DATAFILE "Add1.Dat" OF AddRec;
     ShAdd  :     AddRec;
ENDEXTERNAL;
```

Once the respondent's address details have been taken during the interview, the READFILE command is used to search the external file and, if an existing entry is found, routes to a special notification question. The search itself typically takes approximately one second, which is impressive by any standards.

2.4 Data transfer for analysis

Blaise has its own tabulation package, Abacus, which is ideal for producing very quick counts and percentages on your data. However, it cannot produce the wide range of statistics available with packages such as SPSS/PC+ and SAS. Data can be transfered to other packages by performing two steps.

(i) Converting Blaise data to ASCII
(ii) Using the setup generator

The conversion utility offers a great deal of flexibility in that fixed or free format files can be generated, using a variety of delimiters. The other advantage is that the settings can be stored in a file and run from the command line.

The setup generator is a facility consisting of a number of programs which read information from a Blaise database and write out the commands, labels etc necessary to read the data into another software package. Once the program has been generated, it can be executed in the 'other' package.

In the case of our example, the conversion program is run from a batch file and then an SPSS/PC+ data list program (created by the setup generator) is run to read in the resultant ASCII file. From here a whole range of standard analyses can be performed, and because it can all be done with batch files, all the interviewer has to do is select an option on the menu.

2.5 Letter generation

Very often after a survey has been conducted, there may be a need to follow-up some or all of the respondents with a letter containing further information. With a paper based survey, this would probably take the form of a standard letter without any form of personalisation except the address label. Having a computer based survey opens up a world of possibilities because the data can be easily transfered from one piece of software to another.

For the example in hand, firstly the data is converted into an ASCII dataset, using the conversion utility, as was done before transfering to SPSS/PC+. Next, a Fortran program is used to take the ASCII file and then:

(i) Select only those cases for which letters are required, based upon a filter question
(ii) Generate new text fields, based upon questionnaire responses, for text substitution in the letters
(iii) Create a new, comma delimited ASCII file

The resultant ASCII file can now be accessed by the word processing software, in our case Microsoft word. With Microsoft Word, a (semi) standard letter can be written with gaps for not only name and address but for absolutely any part of the letter. These gaps are filled with fieldnames which cross-reference the new ASCII file for each individual letter. In this way, if the letter is carefully constructed, it can be personalised to such an extent that it makes references to various things that respondents have said during the interview.

The beauty of this solution is that although only one letter is written, numerous versions are created by the link to the Blaise data. As with the other aspects of this system, the majority of the functions can be performed by a batch file, however the actual "merging" of data and letter has to take place from within Microsoft Word.

3. Summary

This system was designed to give an indication of what can be done with the Blaise utilities and other software which you may already be using. The important point that I am trying to convey here is that although the Blaise interview program is robust and effective, there are other aspects within the system, eg. Abacus and Manipula, which should be used to their full potential. Manipula in particular is immensely flexible but almost certainly under-utilised. Making full use of these tools will mean investing time to learn about them, above and beyond the time usually spent on a survey, but I guarantee it will be time well spent.