Poster Session

Role of Manipula Programs in Support of a Blaise III v1.05 Institutional CATI Study

Patrick Murphy, Battelle/Survey Research Associates USA

1. Overview of CATI Study

The Business Responds to AIDS Benchmark Study (BRTA) is funded by the U.S. Centers for Disease Control and Prevention (CDC). Battelle/SRA is responsible for collecting data from companies in the United States. Participation is voluntary. There are five primary objectives of this data collection effort: (1) to provide a baseline with which trends in business activity regarding HIV/AIDS prevention and education can be tracked; (2) to determine the nature and extent of current AIDS workplace policies and education programs; (3) to estimate the overall impact of the Business Responds to AIDS program which was initiated by the Office of HIV/AIDS at CDC in December 1992; (4) to determine the adoption and diffusion of the major components of the Business Responds to AIDS program; and (5) to understand the factors that influence the decision to adopt and implement an AIDS workplace policy, educational program, or philanthropic effort.

Battelle/SRA assisted the CDC investigators in the development and testing of the survey questionnaire which includes questions on a company's demographics, worksite policies, education programs, and philanthropic activities. The survey questionnaire was programmed into a computer assisted telephone interview system (CATI) using Blaise III by Battelle/SRA staff. The CATI system consists of a call scheduling system and a data entry program. On average, the CATI interview takes about 20 minutes to complete. An average of six to eight interviewers simultaneously access the CATI system using a Novell network. Approximately 4,000 firms will be called. To date, the interview has been administered in English, Spanish, and Cantonese.

The BRTA project is an institutional CATI survey which contacts businesses Monday through Friday from 9AM to 5PM local time. The time zone field feature of the Blaise III call scheduler was implemented in this study to compensate for time zone differences across the United States. The survey is divided into eight sections. If the initial respondent cannot provide answers for a section, s/he is encouraged to provide a reference to another person at their company who may be able to answer those sections. Therefore, there may be more than one respondent per firm. To manage this fact, a dual key is assigned to each case, FIRM_ID and PERS_ID, to identify the firm and the person responding to the questions, respectively. The initial respondent is assigned a PERS_ID of '1' and subsequent respondents from the same company are given PERS_IDs of '2', '3', etc. When a respondent gives a referral, the first respondent's data are removed from the main Blaise data base and are saved in another area. A blank FORM is inserted into the main Blaise data base for the referral person. (The term "FORM" is Blaise III terminology for a unique record in the primary study data base. For this study, the primary data base is called CATIVER.~BD.) In this way, only one person from

each firm is in the Blaise data base at any given time, ensuring that only one person per firm will be included in the day batch by the Blaise call scheduler. Another criterion specified by the client was that only one person per firm is allowed to answer a section of questions. Therefore, a method was devised to pass information from the previous respondent's FORM to the current respondent's FORM regarding which sections are still available to be answered by the current respondent.

The multiple respondent feature of the CATI system, customized appointment setting features, and a customized refusal conversion system were all implemented using Manipula programs. This paper describes the structure of the CATI system that Battelle/SRA developed and how Manipula programs were used to implement the study requirements.

2. Structure of CATI Questionnaire System

The CATI questionnaire system consists of three Blaise program files: two questionnaire modules and one CATI module for controlling the flow of logic through the questionnaire and for establishing the CATI blocks.

The two Blaise questionnaire modules are BRTABLOK.BLA and ANSTYPES.BLA. The BRTABLOK.BLA file contains the study question texts in BLOCK format. The RULES sections for each of the BLOCKs are included in this file. The ANSTYPES.BLA file contains answer types for enumerated questions such as (YES, NO) and (HUMAN RESOURCES, FINANCES, CORPORATE POLICY).

The Blaise CATI module, CATIVER.BLA, performs several functions. First, it initializes the system as a CATI system by telling Blaise III to use CATI scheduling data structures. Second, it establishes CATI blocks such as NONRESPONSE and APPOINTMENT. A special block, BIDENT, contains identification information (e.g., name, phone number, time zone, comments) on the respondents. It also has a field that indicates which questionnaire sections are available for the respondent to answer. Third, it contains code to address the special requirements of this study. The most important of these requirements is that each of the sections of questions contained in BRTABLOK.BLA is answered only once by each firm. An IF-THEN-ENDIF logical construct surrounds each section of questions to ensure that each section of questions is only answered once. A further requirement of this study is that multiple people from the same company are allowed to answer different sections (BLOCKs) of the questionnaire but the same section cannot be answered by more than one person. This was accomplished by passing a variable to a new FORM identifying the sections that were unanswered in the old FORM. The IF-THEN-ENDIF logical construct surrounding each BLOCK of questions evaluates this variable to make sure that the BLOCK has not been answered by a previous person at the company before it allows another person into the BLOCK. This logic ensures that a BLOCK of questions is only answered once by any one person.

3. Structure of the Manipula Program System

Manipula programs are used to manipulate data as they enter the CATI system, as they reside in the system, and as they exit the system. For the Business Responds to AIDS project, the majority of Manipula programs are interfaced with the Blaise III call scheduler.

Each subsection that follows describes how Manipula was used in support of this project.

Sample data

A Manipula program was written to load the sample data into the primary study data file, CATIVER.~BD. The program that performed this function is SAMPLE.MAN.

SAMPLE.MAN A block titled "BIDENT", containing the identifying information for each member of the sample, was used as a "DATAMODEL" for the input data in SAMPLE.MAN. The data structure in BIDENT was modeled after the structure of sample data provided by the client. An ASCII file called SAMPLE.DAT contained the sample information formatted according to the BIDENT data structure.

Sample data are read into the Blaise data file, CATIVER.~BD, by using simple INFPUTILE and OUTPUTFILE statements pointing to the BIDENT data model and the CATIVER data models, respectively. An initial appointment of "Monday through Friday, 9AM to 5PM local time" is made for each sample member. The appointment is made in the "Manipulate" section by assigning values to the CATI data structure "catimana.catiappoint" fields.

Referral cases

A special requirement of the BRTA project is to allow more than one person from a company to answer the survey sections. Additionally, only one person from a company is allowed to answer any one section. For example, assume John Doe from Acme Manufacturing Company answered Sections A, C, and E of the questionnaire. He recommended that the personnel director, Jane Doe, answer Sections B, D, and F. She must not be allowed to answer Sections A, C, or E because the first respondent already did.

Before each section of the questionnaire is asked, the CATIVER.BLA program checks to see if that section has been answered by a previous respondent. If not, the questions in that section are routed to the screen for the interviewer to see. The first question in each questionnaire section is "Can you answer questions regarding [the subject of this section]?" If the respondent says yes, then the rest of the questions in the section are routed to the screen. At the end of each questionnaire section, there is a question that states "This section is done. Press 1 to continue." When the interviewer presses '1', that section is marked as complete and is added to the variable which lists the completed sections for that firm. If the respondent states that s/he cannot answer questions contained in that particular section, then the CATIVER.BLA program evaluates the next section of questions. At the end of the interview,

the respondents are asked to provide a the name of a person at their company who can answer the questions in the sections that have been skipped. Because the end of the interview has been reached, the DIALRESULT for this attempt call is 1 (Completed Interview).

SPAWN.MAN The Manipula program SPAWN.MAN scans each FORM in CATIVER.~BD to see if the interview is complete ("complete" is defined as "the respondent has been asked to complete all the sections, even if all the sections were not answered") and if the respondent gave a reference to another person at their company. If these two conditions are met, the FORM enters the "spawning" process. The first step of the "spawning" process is for the referral person's identifying data to be written to an ASCII file called SPAWN.DAT. These identifying data follow the structure defined in the BIDENT block, including the name, title, telephone number, and any specific comments about the referral person. Because a new FORM will be generated for these data, a new set of key data (FIRM_ID and PERS_ID) items must be generated for the referral person. The FIRM_ID for the referral person remains the same but the PERS_ID is incremented by one. For example, if the first interview was with FIRM_ID=10022 and PERS_ID=1, then the referral record would have the FIRM_ID=10022 and PERS_ID=2.

SAVEREC.MAN After the SPAWN.MAN program is run, a Manipula program called SAVEREC.MAN is executed. The purpose of this program is to remove the FORMs from CATIVER.~BD for all cases in which a new case is being "spawned". The SAVEREC.MAN program searches CATIVER.~BD for cases in which the interview is complete (DIALRESULT=1) and a reference was given. (These are the same criteria that the SPAWN.MAN program uses for creating a record for SPAWN.DAT.) These FORMs are moved from CATIVER.~BD to a data file called SAVEREC.~BD which has the same format as CATIVER.~BD. The purpose of this procedure is to ensure that only one FORM per firm will be in CATIVER.~BD at any given time.

ADDREC.MAN The last steps in the "spawning" process occur in the Manipula program ADDREC.MAN. The data in the ASCII file SPAWN.DAT are used to create new FORMs in CATIVER.~BD. The program reads the data in SPAWN.DAT, using the BIDENT block information as a DATAMODEL, and writes them to CATIVER.~BD.

Summary To summarize the "spawning" process, data for referral respondents are written to an ASCII file called SPAWN.DAT using the BIDENT block as a DATAMODEL. A list of the unanswered sections for that FIRM_ID is also written in SPAWN.DAT. The existing data for firms which gave referrals are copied to SAVEREC.~BD, a Blaise data file with the same structure as CATIVER.~BD. New forms, with the same FIRM_ID and an incremented PERS_ID, are written to CATIVER.~BD using SPAWN.DAT as the source of input data.

Refusal cases

For this study, the BRTA coordinator wanted only one interviewer to conduct refusal conversions. Additionally, he wanted all calling statistics on refusal conversion to be kept separate from all other calling statistics. These criteria were met by setting up a separate area

on the network for refusal conversions. FORMs for respondents who refused to participate were moved from the main CATIVER.~BD data base to the REFUSAL.~BD data base. This process of moving the FORMs and performing other necessary details was carried out by the Manipula program REFUSAL.MAN.

REFUSAL.MAN REFUSAL.MAN was executed to identify and extract FORMs from CATIVER.~BD whose respondents refused to participate. If the last dial result for the FORM was '5' (REFUSAL), the FORM is moved from CATIVER.~BD to the REFUSAL.~BD data file in another subdirectory. The REFUSAL.~BD data file has the same format as the CATIVER.~BD data file. The REFUSAL.MAN program changes the last dial result from '5' (REFUSAL) to '4' (APPOINTMENT) for each FORM so that the call scheduler will include those cases in the day batch. If the last dial result is left at '5', the Blaise call scheduler considers the case 'done' and will not include the case in a day batch. The study manager chose to set a period appointment for these cases, Monday through Friday, 9 AM to 5PM local time. All of the calling statistics will be kept in the REFUSAL.~TH file in this subdirectory, allowing separate calling statistics to be maintained.

Not yet called cases

FORMs are loaded into the CATIVER.~BD data base 500 at a time by the program SAMPLE.MAN. The day batch size used for this study is 200. An average of six to eight interviewers spend approximately six hours each per business day going through the day batch. The result of these parameters is that not all the FORMs get called over a two to three week period. Once a case is started, the appointments set for it are given higher priority (by the call scheduler system) than cases not contacted. Approximately three weeks after a sample of 500 cases is loaded, a significant portion of these cases may not have been contacted. A Manipula program NOCALL.MAN was written to resolve this problem.

NOCALL.MAN The Manipula program NOCALL.MAN scans CATIVER.~BD for cases in which the CATI variable "number of calls" is zero. For cases with "number of calls" equal to zero, the program makes a period appointment for that case from 9AM to 5PM the next day. This will change the priority of the case from default (0) to medium (2).

Appointmentexpired cases

If a case has a period appointment (e.g., Monday through Friday, 9AM to 5PM) and the case is never called during that period, the call scheduler priority for that case drops to default, the lowest priority. For the purposes of the BRTA study, a case which has not been called during its appointment period should have a higher priority than the original appointment, not a lower priority. This was accomplished by a Manipula program called NODEFAULT.MAN.

NODEFAULT.MAN The Manipula program NODEFAULT.MAN scans the CATIVER.~BD data file for cases which have an expired appointment period. If a case had a period appointment and the DATEEND of the appointment had passed, then the program makes

another appointment for the case for the next day, 9AM to 5PM local time. This changes the call scheduler priority of the case from default (0) to medium (2).

3. Conclusion

The Manipula programming language, provided with Blaise III and integrated into the system interface, is a powerful tool for customizing the Blaise III call scheduler to the particular needs of a project. The BRTA project used Manipula programs to:

- load sample data,
- "spawn" new FORMs,
- segregate refusal FORMs,
- make appointments for firms that haven't been called, and
- make appointments for firms with expired appointments.

By gaining skill with the Manipula programming language, the CATI programmer can say "Yes" to many more requests from CATI study managers.

Appendix

```
{
       SAMPLE.MAN
       ADD SAMPLE DATA TO DATA FILE
}


uses brtacati "cativer"

   datamodel address
      block bident
         fields
            firm_id: integer[5]
            pers_id: integer[2]
            firm_name : string[30]
            firm_phone: string[14]
            contact : string[25]    {firstname+lastname}
            address : string[30]
            city    : string[16]
            state   : string[2]
            zipcode : string[5]
            timezone : string[3]
            sic      : string[6]
            industry : integer[1]
            empsize  : string[1]
            size     : integer[1]
            abl      : integer[1]
            abinum   : string[9]
            employ   : integer[5]
            title   : string[20]
            location: string[30]
            sectflag: string[16]
            sectdone :array[1..8] of integer[1]
            callername: string[16]
            refcmnt: string[80]
      endblock

      fields
```

```
         ident : bident
   endmodel

inputfile
   infile: address ('SAMPLE.DAT', ascii)

outputfile
   outfile: brtacati ('cativer', blaise3)
   settings makenewfile = no

manipulate
   outfile.catimana.catiappoint.appointtype := 3
   outfile.catimana.catiappoint.datestart := sysdate
   outfile.catimana.catiappoint.timestart := totime(09,00,0)
   outfile.catimana.catiappoint.dateend   := sysdate+7
   outfile.catimana.catiappoint.timeend   := totime(17,00,0)
   outfile.catimana.catiappoint.whomade   := 'WAVE4'
   write(outfile)
```

```
{
       SPAWN.MAN
       FORMATTED FOR CATIVER   FINAL VERSION
       GENERATE NEW CASE, FOR REFERRAL
 }

uses cativer 'cativer'

   datamodel address

      block bident
         fields
            firm_id: integer[5]
            pers_id: integer[2]
            firm_name : string[30]
            firm_phone: string[14]
            contact : string[25]    {firstname+lastname}
            address : string[30]
            city    : string[16]
            state   : string[2]
            zipcode : string[5]
            timezone : string[3]
            sic      : string[6]
            industry : integer[1]
            empsize  : string[1]
            size     : integer[1]
            abl      : integer[1]
            abinum   : string[9]
            employ   : integer[5]
            title   : string[20]
            location: string[30]
            sectflag: string[16]
            sectdone :array[1..8] of integer[1]
            callername: string[16]
            refcmnt   : string[80]
      endblock

      fields
         ident : bident
```

```
   endmodel

inputfile
   infile  : cativer ('cativer', blaise3)

outputfile
   outfile : address ('spawn.dat', ascii)

manipulate   { cases with interview dial result ;
               the interview is done for this one}

   if catimana.caticall.regscalls[1].dialresult = 1 then
         if infile.blkheard.refask=1 then       {gave reference}
            outfile.ident.pers_id := ident.Pers_id + 1
            outfile.ident.callername := 'SPAWNED'
            outfile.ident.contact  :=   infile.blkheard.refname
            outfile.ident.title    :=   infile.blkheard.reftitle
            outfile.ident.location :=   infile.blkheard.refdept
            outfile.ident.firm_phone := infile.blkheard.reftel
            outfile.ident.refcmnt :=    infile.blkheard.refcmnt
            write(outfile)
         endif
   endif
{
 SAVEREC.MAN
 FINAL VERSION
 delete case from cativer database
 save case to saverec database
}

uses cativer 'cativer'

updatefile            {update because we want to read and write}
   updtfile  : cativer ('cativer', blaise3)

outputfile
   saverecs : cativer ('saverec', blaise3)
   settings makenewfile = no
```

```
    manipulate

       { cases with interview dial result }

 if (catimana.caticall.regscalls[1].dialresult = 1) AND
(blkheard.refask=1) then
       write(saverecs)
       delete(updtfile)
    else
       write(updtfile)
 endif
```

```
{
  ADDREC.MAN
  final version
  add spawned records in tmpspawn system to cativer system
}
uses cativer 'cativer'

   datamodel address

      block bident
         fields
            firm_id: integer[5]
            pers_id: integer[2]
            firm_name : string[30]
            firm_phone: string[14]
            contact : string[25]    {firstname+lastname}
            address : string[30]
            city    : string[16]
            state   : string[2]
            zipcode : string[5]
            timezone : string[3]
            sic      : string[6]
            industry : integer[1]
            empsize  : string[1]
            size     : integer[1]
            abl      : integer[1]
            abinum   : string[9]
            employ   : integer[5]
            title   : string[20]
            location: string[30]
            sectflag: string[16]
            sectdone :array[1..8] of integer[1]
            callername: string[16]
            refcmnt:   string[80]
      endblock

      fields
         ident : bident
   endmodel
```

```
inputfile
   infile : address ('spawn.dat', ascii)

outputfile
   outfile  : cativer ('cativer',blaise3)

   settings makenewfile = no

manipulate
   outfile.catimana.catiappoint.appointtype := 3
   outfile.catimana.catiappoint.datestart := sysdate
   outfile.catimana.catiappoint.timestart := totime(09,00,0)
   outfile.catimana.catiappoint.dateend   := sysdate+7
   outfile.catimana.catiappoint.timeend   := totime(16,00,0)
   outfile.catimana.catiappoint.whomade   := 'SPAWNED'

   write(outfile)
```

```
{
       REFUSAL.MAN
       FORMATTED FOR CATIVER, FINAL VERSION
       COPY REFUSAL RECORDS TO SEPARATE SUBDIRECTORY
}
uses cativer 'cativer'

   datamodel address

      block bident
         fields
            firm_id: integer[5]
            pers_id: integer[2]
            firm_name : string[30]
            firm_phone: string[14]
            contact : string[25]    {firstname+lastname}
            address : string[30]
            city    : string[16]
            state   : string[2]
            zipcode : string[5]
            timezone : string[3]
            sic      : string[6]
            industry : integer[1]
            empsize  : string[1]
            size     : integer[1]
            abl      : integer[1]
            abinum   : string[9]
            employ   : integer[5]
            title   : string[20]
            location: string[30]
            sectflag: string[16]
            sectdone :array[1..8] of integer[1]
            callername: string[16]
            refcmnt: string[80]
      endblock

      fields
         ident : bident
   endmodel
```

```
     updatefile
        updtfile  : cativer ('cativer', blaise3)

     outputfile
        outfile : cativer ('k:\cai\brta\refusal\refusal', blaise3)
        settings makenewfile = no

     manipulate   { cases with refusal dial result ;
                    the interview is done for this one}
        if catimana.caticall.regscalls[1].dialresult = 5 then
           outfile.catimana.caticall.regscalls[1].dialresult := 4
           outfile.catimana.catiappoint.appointtype := 3
           outfile.catimana.catiappoint.datestart := sysdate
           outfile.catimana.catiappoint.timestart := totime(09,00,0)
           outfile.catimana.catiappoint.dateend   := sysdate+
           outfile.catimana.catiappoint.timeend   := totime(1,00,0)
           outfile.catimana.catiappoint.whomade   := 'Refusal'
           write(outfile)
           delete(updtfile)
        else
           write(updtfile)
        endif
     {
       NODEFAULT.MAN
       FINAL VERSION
       IF AN APPT PERIOD IS OVER, MAKE NEW APPT SO CASE WON'T BE
DEFAULT
     }

     uses cativer 'cativer'

        datamodel address

           block bident
              fields
                 firm_id: integer[5]
                 pers_id: integer[2]
                 firm_name : string[30]
```

```
            firm_phone: string[14]
            contact : string[25]    {firstname+lastname}
            address : string[30]
            city    : string[16]
            state   : string[2]
            zipcode : string[5]
            timezone : string[3]
            sic      : string[6]
            industry : integer[1]
            empsize  : string[1]
            size     : integer[1]
            abl      : integer[1]
            abinum   : string[9]
            employ   : integer[5]
            title   : string[20]
            location: string[30]
            sectflag: string[16]
            sectdone :array[1..8] of integer[1]
            callername: string[16]
            refcmnt:   string[80]
      endblock

      fields
         ident : bident
   endmodel

updatefile
   updtfile  : cativer ('cativer',blaise3)

manipulate
 IF updtfile.catimana.catiappoint.appointtype = 3 THEN
   IF updtfile.catimana.catiappoint.dateend <= sysdate THEN
      updtfile.catimana.catiappoint.appointtype := 3
      updtfile.catimana.catiappoint.datestart := sysdate+1
      updtfile.catimana.catiappoint.timestart := totime(09,00,0)
      updtfile.catimana.catiappoint.dateend   := sysdate+1
      updtfile.catimana.catiappoint.timeend   := totime(17,00,0)
      updtfile.catimana.catiappoint.whomade   := 'no deflt'
   ENDIF
```

```
ENDIF
 write(updtfile)
```

```
{
  NOCALL.MAN
  final version
  make medium appointments for cases with 0 calls
}

uses cativer 'cativer'

   datamodel address

      block bident
         fields
            firm_id: integer[5]
            pers_id: integer[2]
            firm_name : string[30]
            firm_phone: string[14]
            contact : string[25]    {firstname+lastname}
            address : string[30]
            city    : string[16]
            state   : string[2]
            zipcode : string[5]
            timezone : string[3]
            sic      : string[6]
            industry : integer[1]
            empsize  : string[1]
            size     : integer[1]
            abl      : integer[1]
            abinum   : string[9]
            employ   : integer[5]
            title   : string[20]
            location: string[30]
            sectflag: string[16]
            sectdone :array[1..8] of integer[1]
            callername: string[16]
            refcmnt:   string[80]
      endblock

      fields
         ident : bident
```

```
   endmodel

 updatefile
   updtfile  : cativer ('cativer',blaise3)

 manipulate
   IF updtfile.catimana.caticall.nrofcall=0 then
      updtfile.catimana.catiappoint.appointtype := 3
      updtfile.catimana.catiappoint.datestart := sysdate+1
      updtfile.catimana.catiappoint.timestart := totime(09,00,0)
      updtfile.catimana.catiappoint.dateend   := sysdate+1
      updtfile.catimana.catiappoint.timeend   := totime(17,00,0)
      updtfile.catimana.catiappoint.whomade   := 'NOCALL'
   ENDIF
write(updtfile)
```