

## **Pretesting Interactive Voice Response / Automated Speech Recognition Surveys**

Sid J. Schneider  
David Cantor  
Tracey Hagerty Heller  
Pat Dean Brick

Westat  
Rockville, Maryland

This paper was presented at the International Conference on Questionnaire Development, Evaluation, and Testing Methods, Charleston, South Carolina, November 14-17, 2002.

Direct inquiries to Sid J. Schneider, Westat, 1650 Research Boulevard, Rockville, MD 20850, or email [sidschneider@westat.com](mailto:sidschneider@westat.com).

In an Interactive Voice Response (IVR) survey, a “talking computer” conducts interviews by playing digitized voice files over the telephone. The computer emulates a human interviewer, presenting the instructions for the survey, the questions, and the alternative responses. When the IVR survey uses Touch Tone Data Entry (TDE), the respondent replies to the questions by pressing the buttons on the telephone keypad, such as by pressing 1 for “yes” or 2 for “no.” When the IVR survey uses Automated Speech Recognition (ASR), the respondent replies to the questions by speaking, such as by saying “Yes” or “No” over the telephone. The ASR software digitizes the spoken response and extracts the respondent’s answer.

Contemporary ASR systems are far more limited than human listeners. The ASR software typically checks a vocabulary list of the possible words and phrases that the respondent might say at a particular point in the interaction. For example, for a yes-or-no question, the vocabulary would include “yes,” “no,” and all equivalent words and phrases like “not at all.” The software assigns a number to each alternative, corresponding to the level of confidence that the alternative is the one that the respondent actually said. When the level of confidence for an alternative exceeds a pre-selected amount, the software selects that alternative as the response.

IVR/ASR surveys are currently uncommon, largely because of the limited capabilities of ASR systems. Any IVR/ASR survey that is fielded today requires rigorous pretesting to help ensure that respondents provide quality data and are satisfied with the interview process.

The purpose of this paper is to describe current IVR/ASR systems and the issues that surround surveys using this technology. To illustrate these issues, the paper presents an example -- an IVR/ASR system for collecting Census 2000 short form data. The final sections provide an overview of the steps needed to develop and test an IVR/ASR survey.

### **Current IVR/ASR systems**

Most current IVR/ASR applications are not intended for survey data collection. They are call routing, account inquiry, or customer assistance systems designed to connect callers to a requested telephone extension, or to dispense brief items of information like stock prices, account balances, or the time a flight is scheduled to arrive. Surveys are very different from these common IVR/ASR applications.

With a call routing or stock quotation systems, callers interact with the IVR/ASR system for only a few minutes, to achieve one or two goals. In an IVR/ASR survey, respondents interact with the system over a much longer period of time. The respondents may have to answer many questions over perhaps ten or twenty minutes. Designers of IVR/ASR survey systems therefore must ensure that their systems do not sound monotonous and drawn out, even when they present repetitive questions, such as successive yes-or-no questions.

Unlike IVR/ASR surveys, most current IVR/ASR systems are designed to be used on many occasions. For example, a caller might use the system almost every day to check the value

of a mutual fund. For that reason, the systems often have a “speak ahead” feature in which the callers are not required to listen to each question in its entirety. Callers who remember a question from an earlier call can interrupt the question with the reply. By contrast, survey respondents usually must hear each question completely. IVR/ASR surveys therefore usually cannot have a “speak ahead” feature and may seem artificial and regimented. Researchers must ensure that their surveys do not make respondents impatient.

In most IVR/ASR applications, callers seldom need to skip a question or return to an earlier question. In surveys, however, respondents often wish to do these things. In self-administered paper-and-pencil surveys or in CATI surveys, respondents can skip questions or return to earlier questions with minimal effort. An IVR/ASR survey, however, cannot easily make these actions available to the respondents.

Most importantly, surveys often deal with complex concepts. A caller to an IVR/ASR system that provides stock quotes or train departure times usually does not need to think through the ambiguities of the questions posed by the computer. By contrast, survey questions can be puzzling. Respondents cannot easily ask for clarification from an IVR/ASR system, because current ASR software cannot accommodate unanticipated questions from respondents. Researchers must be certain that respondents do not find this situation to be exasperating.

In sum, IVR/ASR surveys may be harder to design than more common IVR/ASR applications. Several authors have published guidelines for the design of common IVR/ASR applications (e.g., Edgar, 1997; Gardner-Bonneau, 1999; Schumacher, Hardzinski & Schwartz, 1995), but those guidelines may not necessarily apply to IVR/ASR surveys. For example, the Human Factors and Ergonomics Society (2001) has released guidelines for the design of IVR systems, including IVR/ASR systems. These guidelines are relevant to applications like customer service lines, but they do not consider the special requirements of surveys. The optimum guidelines for IVR/ASR surveys are yet to be agreed upon.

### **Advantages of IVR/ASR surveys**

Despite the challenges of designing them, IVR/ASR surveys offer a number of potential advantages:

**Economy.** The most obvious potential advantage of IVR/ASR surveys is their cost. The price of voice boards, ASR software, and programming is relatively high. Once an IVR/ASR system is operating, however, survey data can be collected with relatively little human intervention. IVR/ASR surveys are usually inbound systems, that receive calls from the respondents. Some are outbound systems, in which a human interviewer calls a respondent and then transfers the call to the IVR/ASR survey. In either case, an IVR/ASR system could be a much cheaper telephone-based data collection strategy than computer-assisted telephone interviews (CATI) conducted entirely by a human interviewer. These potential cost savings motivated the Bureau of Labor Statistics (BLS) to try an IVR/ASR system for their Current Employment Statistics (CES) survey. The BLS’ experience with CES confirmed that the

IVR/ASR method brought higher capital costs but lower labor costs, as compared with CATI (Clayton & Winter, 1992).

**Language and Technology.** IVR/ASR surveys, like CATI surveys, can collect data from respondents with limited literacy skills who cannot complete paper-and-pencil questionnaires. The systems can present voice files in foreign languages, to reach foreign language speakers without the cost of foreign language interviewers.

Unlike IVR/TDE surveys, IVR/ASR surveys can include households that lack touch tone telephones. In North America, the vast majority of telephones use touch tone technology, but in Europe, rotary dial telephones and button telephones that use pulses are still common (Blyth, 1998; Gardner-Bonneau, 1999). Even in the United States, lower-income households disproportionately lack touch tone telephones. This disparity led the BLS to build an IVR/ASR system rather than an IVR/TDE system for their recent CES survey.

**Ease of use relative to TDE.** Touch tone data entry is awkward with cellular telephones and other telephones in which the keypad is on the headset, while spoken data entry is easy with any sort of telephone. Also, since IVR/ASR systems attempt to emulate conversation, they may seem more “natural” and pleasant to use than TDE systems. In one study (Clayton & Winter, 1992), 60 percent of the respondents preferred ASR to TDE.

**Perceived privacy.** Respondents tend to view computer-assisted data collection systems as private and confidential (Turner, Ku, Rogers, Lindberg, Plaeck, & Sonenstein, 1998; Tourangeau & Smith, 1998). IVR/ASR systems may also be perceived that way, because they ask respondents to share their information with an automated system rather than another person (Cooley, Miller, Gribble & Turner, 2000). They may therefore provide an effective way to collect sensitive personal data (Mingay, 2000). Tourangeau, Steigler, and Wilson (2002) found that an IVR/ASR survey obtains more honest answers to satisfaction questions, as compared with a CATI survey.

**Standardization.** Unlike CATI interviewers, IVR/ASR systems always present the questions and instructions in a standardized, unvarying way. This standardization may reduce measurement error in an IVR/ASR survey. On the other hand, standardization may not always be desirable. CATI interviewers may sometimes collect better quality data by tailoring their presentation to the characteristics of individual respondents (Dijkstra & Smit, 2002; Schaeffer, 1991).

Because of these potential advantages, IVR/ASR surveys may well become more common as ASR technology improves. At present, no one can have a truly natural-sounding conversation with an ASR system because of its imperfect ability to “understand” human speech. As ASR software improves, the advantages of IVR/ASR surveys may increasingly outweigh that drawback.

## **Rationale for testing an IVR/ASR survey**

Developers of IVR/ASR surveys cannot rely exclusively on their own judgment to find the best possible designs. They must pretest their systems and modify their systems on the basis of their findings. The purpose of pretesting is to avoid potential problems in an IVR/ASR survey by letting users' opinions and observations guide the development of the system. This user-centered approach ideally leads to a survey which collects high quality data and leaves the respondents feeling satisfied, rather than like they have been struggling to get through the survey.

IVR/ASR are simultaneously spoken surveys—like person-to-person CATI surveys—and self-administered computer-assisted surveys. IVR/ASR surveys must be tested both as spoken surveys, for the wording of the questions and the respondents' comprehension, and as computer-assisted self-interviews, for the respondents' ability to navigate through the survey without direct assistance from a human interviewer. Researchers have developed methods for pretesting spoken surveys and for pretesting interactive software. By combining these methods, developers of IVR/ASR surveys can ensure that their products are not only well-accepted by respondents but also effective data collection tools.

Kamm (1994) summarized the design goals for an IVR/ASR system by posing these two questions:

Would a typical user judge the automated interaction to be satisfactory--that is, is the user successful at completing the task, and is the interaction easy and efficient from the user's viewpoint?

Is the degree of successful automation sufficient to make its use financially attractive for the application provider? Do the benefits and savings accrued from the automation effort offset the costs of implementation?

In the case of IVR/ASR surveys, a third question is appropriate: Are the collected data of high enough quality to achieve the survey goals?

## **Methods for testing IVR/ASR applications**

Several writers have described methods for testing IVR/ASR applications. Weinschenk and Barker (2000) described five stages to testing an IVR/ASR system:

**Investigation.** The developers review existing IVR/ASR applications, assess the limits and advantages of various approaches, and create a development plan.

**Analysis.** The developers conduct an audience analysis, in which they identify and describe the potential users of the system. They conduct interviews with some potential users to assess their expectations for the proposed system. The developers also conduct a task analysis,

in which they specify the actions that users must carry out to reach their goals. They specify the usability objectives for the system, which might include rapid task performance and accuracy.

**Conceptual model.** The developers develop scenarios of how various sorts of users, with various goals, will interact with the system. They examine the users' workflow in various scenarios, and revise the system to meet the usability objectives.

**Detail design.** The developers create prototypes of the system using storyboards, sketches, and other design documents. They conduct walk-throughs, in which they emulate callers to the system, having diverse goals. They note the parts of the call when callers might have difficulty, and redesign the system to improve those sections. When the developers are satisfied that the salient problems have been removed from their design, they create the initial version of the software and voice files.

This version undergoes "usability testing," a component of software evaluation in which researchers observe users under controlled conditions (Nielsen, 1993). The usability researchers measure the time required for the users to accomplish specific tasks, identify the tasks that the users have trouble completing, and locate the points at which the users tend to hesitate or become perplexed. The researchers debrief the users after the test to ascertain their reactions to the system. The developers use the results of this testing to revise their system and submit it for another round of usability tests.

**Implementation.** The developers complete the programming and offer the system to the intended users. They obtain user feedback, and adjust the system accordingly.

Markowitz (1996) also listed stages in the development of an IVR/ASR application. Her model contained four stages:

**Wizard of Oz testing.** This testing is analogous to the famous wizard in the movie, who seemed to be performing powerful feats, while actually being an illusion controlled by a man concealed behind a curtain. In Wizard of Oz (WOZ) testing, the application is controlled not by software but by a person who selects the voice files that the user hears. The user is unaware of this situation, much like Dorothy was initially unaware of the man behind the curtain. The advantage of WOZ testing is that no programming is needed to test a prototype. Developers need only the voice files, and a WOZ platform. When the developers want to change their systems for additional testing, they can do so without any programming. WOZ testing helps the developers identify the version of the system that merits further testing.

**Prototype development.** The developers create the software for the selected version of the system.

**Field testing of pilot systems.** Users try out this initial version of the system and report their reactions. These reactions may be collected with questions embedded within the application. Usability testing can occur at this stage of testing.

**Field testing of the complete system.** The developers make the system available to its intended users, who provide feedback that guides further modifications.

Bersen, Dybkjaer, and Dybkjaer (1998) argued that WOZ testing was justified only when the costs of testing a functional, software-based prototype were high. When an IVR/ASR system involves novel or complex interactions, it will probably be costly to program, and WOZ testing may be a cost-effective tool. However, when the system is relatively simple, the developers might benefit from programming a complete system, so that they can observe its actual performance. In addition, some IVR/ASR systems involve retrieving information or performing mathematical calculations. Those systems might be tested best using actual, software-based prototypes because the “man behind the curtain” in a WOZ test may not perform those tasks very quickly.

## **Evaluation Measures**

Testing an IVR/ASR survey can involve obtaining quantitative and qualitative measures.

**Quantitative measures.** The International Committee for Information Technology Standards (2001) has set forth industry-standard measures for software usability evaluations. The standards call for test users to be given tasks to perform with the software. The measures taken include the proportion of tasks that the users were able to complete, and the amount of time required, the amount of help that the users required, and the user’s satisfaction with the system.

These measures are appropriate for testing an IVR/ASR survey. The evaluation can reveal the proportion of users who were able to respond to each question, and the amount of time required. It can reveal the number of recognition failures by the ASR software.

The evaluation can also reveal the quality of the data collected by the survey. Clayton and Winter (1992) obtained an overall error rate in the 1 to 3 percent range for an IVR/ASR survey for the BLS. The errors occurred both when respondents entered data incorrectly, such as by saying “zero” when “blank” was correct, and when the ASR software incorrectly recognized the respondents’ utterances.

**Qualitative measures.** Pretesting an IVR/ASR system can also involve debriefing the users to learn their perceptions on a number of issues including their overall reaction to the system, their ability to understand the computer, the ability of the computer to understand them, and their preference of IVR/ASR over other survey modalities. These data can reveal the correlates of satisfaction with the system. For example, in one study (Boyce, 1999), the users’ overall satisfaction with the survey was more strongly correlated with their perceptions about the amount of time that the survey lasted than with the actual time that it lasted.

## Potential problems in an IVR/ASR survey

The potential problems in IVR/ASR surveys fall into three categories. First are the problems that can affect any spoken questionnaire, such as difficulties with the clarity of the questions or the cognitive burden on the respondents. Second are problems associated with the computer hardware and software, because the computer-mediated “interviewer” is much more limited and error-prone than a human interviewer. Third are problems arising from a mismatch between the expectations of the respondent and the behavior of the system, even when the survey is well-designed and the computer system is functioning adequately.

**Survey design.** An IVR/ASR survey, like any spoken questionnaire, must be designed so that respondents can understand the meaning and purpose of the questions, and the questions do not pose an undue burden. Several excellent articles describe pretesting procedures to avoid those problems (e.g., Forsyth & Lessler, 1991). The current QDET conference will certainly add to that literature.

The computer-mediated format of IVR/ASR surveys, however, poses special challenges for developers as they choose the wording and order of the questions on the survey. For example, an IVR/ASR survey can sound tedious when it presents a succession of questions having the same stem, such as “In the past month, have you or your family...” In an interviewer-administered instrument a common shortcut is for the interviewer to leave out the stem once it is clear that the respondent understands the intent of the question. If there is a pause or side conversation between questions, the interviewer can re-insert the stem once the questioning resumes. For an IVR/ASR application, this anticipation or judgment is not possible. It may be necessary, therefore, for the entire question to be recited for all of the questions, leading to a longer, and perhaps monotonous, instrument.

In addition, developers must choose the alternatives in multiple choice questions carefully. Schumacher, Hardzinski and Schwartz (1995) cautioned that questions on IVR/ASR applications should never have more than four response alternatives. This limitation may be impractical for surveys, however. Survey developers must provide enough response alternatives to collect accurate information, but without placing excessive demands on the respondents’ attention and memory.

**Hardware and software.** ASR software is susceptible to a number of problems:

Failures. Speech recognition failures are a major drawback of contemporary ASR software. Some common survey questions are particularly vulnerable to this problem. For example, ASR software has difficulty recognizing names and dates, so the simple questions “What is your name?” and “What is your birth date?” can result in recognition errors. In one survey for the Census Bureau, the ASR software failed to accurately recognize at least one of these items half of the time (Jenkins & Appel, 1995).

Designers of IVR/ASR surveys can avoid some recognition failures by ensuring that the vocabulary files are complete and contain the regionalisms and slang words that respondents may use. Designers can also avoid speech recognition failures by providing directions about the words that users should employ, such as “Please say ‘yes’ or ‘no.’” For example, an IVR/ASR system for a telephone company achieved only a 55 percent recognition accuracy with the question “Will you accept the charges?” The accuracy rose to 81 percent when the question was changed to “Say ‘yes’ if you will accept the call, otherwise say ‘no’” (Kamm, 1994).

Designers can also avoid recognition failures by including disambiguating questions. For example, a respondent may reply to a question with a number that the ASR software cannot recognize. Of all of the words in the vocabulary, the ASR software may assign the highest confidence levels to “five” and “nine.” In that case, the system might ask “Did you say ‘five,’ ‘nine,’ or another number?” or “Did you say ‘nine,’ yes or no?” These questions can resolve the problem, but they make the interaction quite unlike ordinary person-to-person conversation. If the system asks too many of these disambiguating questions, the respondents may feel frustrated, as though they were trying to communicate with someone who is having trouble hearing them.

Moreover, if the disambiguating questions are not well-designed, the respondent can become trapped in an error loop, in which the respondent says something that the ASR software cannot recognize, the software asks for clarification, but the respondent merely offers another utterance that the software cannot recognize. This situation can sometimes be avoided by wording the disambiguating question in a way that encourages the user to use certain, specific words (Boyce, 1999; Fais, Loken-Kim, & Park, 1995).

Interaction problems. ASR software can be inflexible about the timing of a respondent’s answer. If the respondent speaks before the software is ready to “listen,” the early part of the response may be lost. Some IVR/ASR systems try to avoid this problem by requiring the users to wait for a beep before they answer. Of course, this requirement makes the interaction unlike normal conversation, and burdens the user with an additional task.

Another problem can occur if the respondent takes too long to answer a question, perhaps because the respondent must think about the answer. The ASR software may “conclude” that the respondent is simply remaining silent, and move on to the next question, even though the respondent actually intends to answer. This problem can be avoided by extending the amount of time that respondents have to begin their response. However, if this time is too long, respondents who really do wish to remain silent may be dissatisfied with the long wait.

The timing of the IVR/ASR system’s voice files is also important. The system may seem abrupt if it begins playing a voice file too quickly after the user has finished speaking. However, if the system waits too long before presenting the next voice file, it can seem slow and unresponsive. With the right timing, the system can approximate the flow of normal conversation, even though the usual signals for turn-taking in interpersonal conversation are not available (Karat, Lai, Danis & Wolf, 1999).

All IVR/ASR systems can be thwarted by extraneous noise, such as a dog's bark in the background. The system may erroneously take these sounds as utterances and try to recognize them. The system might then ask the respondent to repeat an answer that the respondent never offered.

Presentation problems. The "personality" of a IVR/ASR system is defined by the behavior of the system. For example, the manner in which the voice of the computer-mediated interviewer greets the respondent and explains the survey at the start of the call helps set the respondent's expectations for the system. The acoustic characteristics of the voice, including its volume, pitch, and intonations, and the variability in these characteristics--sometimes called speech prosody--may influence the respondent's opinions of the survey (McGonagle, 1994). The pace at which the system "speaks" over the telephone may influence the respondents' reactions and ability to understand the system.

Blyth (1997) argued that an IVR/ASR survey should be "as pleasurable as possible," meaning that its behavior must be "humanlike." The respondents should feel that the computer "interviewer" has the same qualities of a good human interviewer. The questions should be asked with the appropriate inflection, with a warm, friendly voice. Cox and Cooper (1981) argued that the best "personality" of an IVR/ASR survey was agreeable and assertive. However, little research exists to support or refute that idea. Conceivably, some respondents find friendliness or assertiveness in a "talking computer" to be distracting or disingenuous. Similarly, IVR/ASR systems probably have to dispense with all small talk and pleasantries, because users may feel uncomfortable "chatting" or "joking" with a computer.

Even a basic decision like the gender of the voice can influence the respondents' reactions to the system. Developers must decide whether a male or female voice is better for a survey. Some developers have asserted that a female voice is better when a survey contains personal questions, but research reports are not unanimous on this issue (Catania, Binson, Canchola, Pollack, Hauck, & Coates, 1996).

Another decision facing developers is whether to use the first person singular in the voice files. Respondents may be disconcerted by a computer that refers to itself as "I," in items like "I will now ask you about your health." On the other hand, a moderate amount of anthropomorphism may make the computer-mediated voice seem more approachable and less impersonal (Gardner-Bonneau, 1999).

Command problems. Human interviewers have no trouble switching from an "interview mode," in which they are asking questions and accepting responses, to a "command mode" in which they are handling requests from the respondent to back up to an earlier question, repeat a question, change an earlier answer, suspend the interview, or modify the presentation, such as by speaking more slowly or loudly. IVR/ASR systems, however, "listen" to the respondent only at specific times, and even then, they listen only for specific words. They are not adept at handling respondent commands or communication about the survey ("meta-communication"; Bersen,

Dybkjaer & Dybkjaer, 1998). As a result, the respondents may feel that they lack control over the progress of the survey (Blyth, 1997; Karat, Lai & Wolf, 1999).

Developers have to find a way around this limitation. For example, they may end the interview with the question, “Would you like to go back and change any of your answers?” and then offer a series of additional questions for those who answer “yes.” Any such solution, however, will fall far short of the flexibility offered by a human interviewer.

Bailout. IVR/ASR systems can provide a method for respondents to access a human interviewer when necessary. For example, if the system cannot recognize a respondent’s utterance after two or three tries, the best solution may be to transfer the respondent to a human interviewer who can complete the survey. Similarly, a respondent who dislikes the computer-mediated format might need a way to transfer the call to a human interviewer.

**User expectations.** Developers cannot aspire to building IVR/ASR surveys with the power of person-to-person surveys, but they can try to build IVR/ASR surveys that live up to the expectations of the respondents. When IVR/ASR surveys do not meet those expectations, respondents will simply hang up. IVR respondents are much more likely than CATI respondents to hang up on the interview, perhaps because IVR respondents could hang up without giving offense to a human interviewer (Cooley, Miller, Gribble, & Turner, 2000; Tourangeau, Steiger and Wilson, 2002). Mingay (2000) found that 2 percent of CATI respondents broke off the interview before the end; the figure for IVR/ASR respondents was 24 percent. IVR/ASR respondents who do not like what they hear will quickly become nonrespondents.

Some respondents may realize that computer technology has its limitations and forgive the shortcomings of an IVR/ASR system. Jenkins and Appel (1995) found that many of their respondents expected and were not greatly bothered by speech recognition failures. Other researchers have found that respondents are not likely to tolerate some of the problems with IVR/ASR systems. For example, respondents may be unable or unwilling to complete IVR/ASR surveys that impose an excessive burden on them (Murphy, Marquis, Hoffman, Saner, Tedesco, Harris, & Roske-Hofstrand, 1999). Respondents often expect surveys to place limited demands upon their memory and attention, and may not tolerate a survey that requires a great deal of mental effort.

A persistent usability problem with IVR/ASR surveys is their slowness. Respondents can conclude that the IVR/ASR technology prolongs the survey, making it unpleasantly time-consuming (Jenkins & Appel, 1995), and much longer than it would be if it were administered by a human interviewer.

Almost no research has been conducted on how survey designers may influence respondent expectations. Should users be told beforehand that the survey will be administered by a computer? Expectations could be quite different if the user knows ahead of time they will be talking to a computer rather than a human. Informing users may set expectations at the appropriate level but reduce the number of respondents who call into the system. User

expectations may also vary by application. For example, expectations could be quite limited in a clinical trials setting, where respondents are volunteers and are asked a predictable set of questions repeatedly over a multiple periods of time. In contrast, expectations could be much higher for respondents to a general household survey who are given an option to call into a toll-free number to participate in a survey.

To a large extent, pretesting an IVR/ASR survey involves assessing user expectations, and continually revising the survey to meet those expectations as much as possible. That is, pretesting involves measuring user expectations, and the fit between those expectations and the realities of the survey. Developers can include texts in an IVR/ASR survey that can shape the users' expectations to some extent. Ultimately, the success of the system depends on the developers' ability to make the interactions as fast and effortless as the users expect.

### **Example: Census 2000 IVR/ASR system**

Schneider, Cantor, Arieira, Malakhoff, Segel, Nguyen, and Guarino (2002) conducted a study in which 5,200 randomly selected households were given the option of providing their Census 2000 short form data with an IVR/ASR system, rather than the usual paper form. Other randomly chosen households were given the option of providing their Census 2000 data on a web-based form, or by telephoning a CATI interviewer. The study suggested that the IVR/ASR survey yielded lower-quality data than did the paper forms, CATI interviews, or web form. The primary reason was that callers to the IVR/ASR survey sometimes hung up midway through the survey. Also, a number of respondents seemed either unprepared to speak at the end of the question, or spoke too slowly to complete their answer. Item nonresponse rates reached 11 to 12 percent for the race and ethnicity items of the IVR/ASR Census 2000 short form. The item nonresponse rates for the other modalities were much lower.

The IVR/ASR survey ended with a brief questionnaire about the respondents' opinions of the survey. Of course, only respondents who reached the end of the survey could complete this questionnaire. Those who hung up on the survey never reached the satisfaction questionnaire. Nonetheless, the results of the questionnaire suggested that respondents' satisfaction with the survey decreased as the amount of time that the survey required increased. Respondents who required a good deal of time to answer the survey, such as respondents with many people in their households or respondents with racially complex households, tended to dislike the survey.

Hispanic respondents were disproportionately likely to report that the IVR/ASR system did not understand them. They also tended to spend more time per question than other respondents.

Respondents with higher numbers of retries for silence tended to rate the ASQ as confusing or frustrating. Retries for silence were occasions when the system repeated a question because the respondent did not reply. Perhaps the respondents' silence reflected their uncertainty about how to answer some of the questions.

Respondents with higher numbers of retries because of invalid responses tended to be more generally dissatisfied with the ASQ, more likely to rate the ASQ as confusing, and more likely to find that the amount of time afforded to respond was inappropriate. These respondents may have been frustrated by the fact that the system repeated questions that they thought they had just answered, when the speech recognition software returned a subthreshold confidence level. Thus, retries tended to diminish the respondents' experience of the ASQ, perhaps by making the survey seem overly long, confusing, unnatural, or unlike human conversation.

These observations are consistent with those of Tourangeau, Steigler, and Wilson (2002) who examined the performance of an IVR/ASR with the Census long form. In this study, some respondents became frustrated with the IVR/ASR and hung up before the interview was over. These respondents tended to come from the larger households.

## **Conclusion**

Developing an IVR/ASR survey is an art that involves working with a very imperfect technology. Developers must ensure that the survey has the characteristics of a well-designed survey, intended to be heard rather than read. They must ensure that the interaction is as effortless as possible, and does not impose undue burden on the respondents. They must ensure that the survey sets the expectations of the respondents realistically, and then lives up to those expectations.

Pretesting an IVR/ASR survey involves potential respondents at every stage, from the conceptualization of the survey, to the development of prototypes, to the programming and implementing the survey. The opinions of potential users guide the developers, in a user-centered design process.

Census Bureau researchers, considering IVR/ASR surveys, wrote "What makes a software user interface usable or unusable? It is usable if it assists users in performing their tasks easily and quickly; if it reflects and understanding of users' goals and tasks; and if the look and feel are consistent and predictable, thus easy to learn and easy to remember...Because people want to minimize or limit the demands on their resources of attention and memory, a design that repeatedly increases those demands beyond the user's preferred level of mental effort will be perceived as difficult and unpleasant to use." (Murphy, Marquis, Hoffman, Saner, Tedesco, Harris & Roske-Hofstrand, 1999). The limitations of contemporary ASR technology increase the demands on respondents. The goal of pretesting is to develop the best possible survey with today's ASR technology, by bringing to bear the expectations and observations of users in the development process.

Pretesting an IVR/ASR survey can be viewed as having these components:

**Cognitive testing.** The survey may be tested like any survey, to ensure that respondents understand the questions and can provide the intended information. One scenario would be to have a human interviewer pretend to be the computer interviewer, to simulate the self-

administered nature of the survey. The interviewer would apply standard cognitive testing techniques.

**Prototype development and expert review.** The survey designers modify the survey on the basis of the results of the cognitive testing. They then plan the IVR/ASR format for the survey, deciding upon such characteristics as the gender of the voice or voices used for the survey, the words and phrases to include in the vocabularies, manner in which the system handles utterances that it cannot recognize with a high level of confidence, the way respondents can change an earlier answer, and so on. The developers prepare a chart depicting the flow of the survey. The development team reviews this chart, possibly with the assistance of outside experts. The team also elicits the opinions of potential respondents, through interviews or focus groups.

At this point, the developers may conclude that the survey is inappropriate for IVR/ASR technology. For example, if the information requirements require a detailed, open-ended set of questions that require probing by an interviewer, a IVR/ASR may not be appropriate. Similarly, if the questionnaire is relatively long and complex, it may also be difficult to effectively implement a design that keeps the respondent interested and motivated.

**Prototype testing.** The developers then decide whether to program a prototype, or to develop a WOZ version of the prototype for testing. In either case, they subject the prototype to testing with potential respondents. They debrief the respondents afterwards to learn their reactions, focussing on factors known to affect IVR/ASR surveys, such as the perceived length of the survey, and the respondents' perceived ability to interact with the system. The prototype may also be tested in a usability laboratory, in which users are observed taking the survey. Usability testing includes measures of the respondents' success in completing the sections of the survey, the amount of time required, and the respondents' satisfaction. The points at which the respondents hesitate or have difficulty are also recorded, to be addressed in the subsequent version of the survey. With new speech application software (VoiceXML), respondents' progress through a survey can be logged and timed automatically as part of usability testing.

Jenkins and Appel (1995) applied this prototype testing procedure when they developed their IVR/ASR survey for the Census Bureau. First, they interviewed potential respondents, to ascertain their preconceived notions about an IVR/ASR survey. Then, they allowed the respondents to complete an early version of the IVR/ASR survey. Afterwards, an evaluator debriefed the respondents. In this way, these developers kept users involved throughout the development process.

**Pilot test.** The survey is pilot-tested in the field. The development team observes the response to the survey, the quality of the data, and elicits the reactions of the respondents through a debriefing interview. Alternatively, the IVR/ASR survey itself can debrief the respondents with questions interspersed throughout the survey and at the end.

## Future Directions

Blyth (1998) thought IVR/ASR technology to be “one of the most exciting – if not the most exciting – for widespread...application.” He noted, however, that the primary problem facing developers is not the accuracy of the software, but the inability to have a “free flowing interview format.”

IVR/ASR surveys are currently rare, and used primarily in market research (Blyth, 1998). However, in the foreseeable future, ASR systems will improve to the point that they will be able to recognize spoken words with an impressive level of accuracy. The Defense Advanced Research Projects Agency currently has two programs to improve speech recognition technology. As these efforts succeed, pretesting will be less concerned with working around the shortcomings of the ASR system, and more concerned with taking full advantage of the computer’s ability to emulate human conversation. At that future time, taking an IVR/ASR survey may be as routine as checking an account balance using an IVR/ASR system is today.

## References

- Bersen, N., Dybkjaer, H. and Dybkjaer, L. (1998) *Designing interactive speech systems: From first ideas to user testing*. London: Springer-Verlag.
- Blyth, B. (1997) Developing a speech recognition application for survey research. In: Lyberg, L.E. Biemer, P., Collins, M., De Leeuw, E., Dippo, C., Schwarz, N., Trewin, D. (eds.) *Survey measurement and process quality*. New York: Wiley.
- Blyth, B. (1998) Current and future technology utilization in European market research. In: Couper, M. Baker, R., Bethlehem, J., Clark, C., Martin, J. Nichols, W., and O'Reilly, J. (eds.) *Computer assisted survey information collection*. New York: Wiley.
- Boyce, S.J. (1999) Spoken natural language dialogue systems: User interface issues for the future. In: Gardner-Bonneau, D. (ed.) *Human factors and voice interactive systems*. Boston: Kluwer Academic.
- Catania, J. A., Binson, D., Canchola, J., Pollack, L. M., Hauck, W. & Coates, T. J. (1996) Interviewer and question effects on sex items. *Public Opinion Quarterly*, 60, 345-375.
- Clayton, R. L. and Winter, D. (1992) Speech data entry: Results of a test of voice recognition for survey data collection. *Journal of Official Statistics*, 8, 377-388.
- Cooley, P., Miller, H.H., Gribble, J.N., and Turner, C.F. (2000) Automating telephone surveys: Using T-ACASI to obtain data on sensitive topics. *Computers in Human Behavior*, 16, 1-11.

- Cox, A.C. and Cooper, M.B. (1981) Selecting a voice for a specified task: The example of telephone announcements. *Language and Speech*, 24, 233-243.
- Dijkstra, W. and Smit, J.H. (2002) Persuading reluctant recipients in telephone surveys. In: Groves, R.M., Dillman, D.A., Eltinge, J.L., and Little, R.J.A. (eds.) *Survey nonresponse*. New York: Wiley.
- Edgar, B. (1997) *PC Telephony*. New York: Flatiron.
- Fais, L., Loken-Kim, K., Park, Y. (1995) Speakers' responses to requests for repetition in a multimedia language processing environment. In: Bunt, H., Beun, R., and Borghuis, T. (Eds.) *Multimodal human-computer communication*. New York: Springer, pp. 264-278.
- Forsyth, B.H., and Lessler, J.T. (1991). Cognitive laboratory methods: A taxonomy. In Biemer, P.P. Groves, R.M. Lyberg, L.E. Mathiowetz, N.A. and Sudman, S. (eds.), *Measurement errors in surveys*. New York: Wiley.
- Gardner-Bonneau, D. (1999) Guidelines for speech-enabled IVR application design. In: Gardner-Bonneau, D. (ed.) *Human factors and voice interactive systems*. Boston: Kluwer Academic, 147-162.
- Human Factors and Ergonomics Society (2001) *HFES-200.5 Human factors engineering of software user interfaces: Interactive voice response (IVR) and telephony*. <http://www.atis.org/pub/IVR/HFES-200-5.pdf> accessed October 1, 2002.
- International Committee for Information Technology Standards (2001) *Common industry standards for usability test reports*. Washington, DC: Author.
- Jenkins, C. and Appel, M. (1995) *Respondents' attitudes towards a U.S. Census voice recognition questionnaire*. Paper presented at the Field Directors' Conference, Orlando, Florida..
- Kamm, C. (1994) User interfaces for voice applications. In: Roe, D.B. and Wilpon, J.G. (eds.), *Voice communication between humans and machines*. Washington, DC: National Academy Press.
- Karat, J., Lai, J., Danis, C. & Wolf, C. (1999) Speech user interface evolution. In: Gardner-Bonneau, D. (ed.) *Human factors and voice interactive systems*. Boston: Kluwer Academic, pp. 1-35.
- Markowitz, J. A. (1996) *Using speech recognition*. Upper Saddle River, NJ: Prentice Hall.

- McGonagle, M. (1994) Voice-recognition and voice-response systems. In: Keyes, J. (ed.) *The McGraw-Hill multimedia handbook*. New York: McGraw-Hill, pp. 35.1 – 35.27.
- Mingay, D. J. (2000) Is telephone audio computer-assisted self-interviewing (T-ACASI) a method whose time has come? *American Statistical Association 2000 Proceedings of the Section on Survey Research Methods*. Alexandria, VA: American Statistical Association.
- Murphy, E., Marquis, K., Hoffman, R. Saner, L., Tedesco, H., Harris, C. and Roske-Hofstrand, R. (1999) *Improving electronic data collection and dissemination through usability testing*. Presented to the Federal Committee on Statistical Methodology. Retrieved on September 30, 2002 at <http://www.fcsm.gov/99papers/emurphy.html>
- Neilsen, J. (1993) *Usability engineering*. Boston: AP Professional.
- Schaeffer, N.C. (1991) Conversation with a purpose--or conversation? Interaction in the standardized interview. In: Biemer, P.P., Groves, R.M., Lyberg, L.E., Mathiowetz, N.A., and Sudman, S. (eds.) *Measurement errors in surveys*. New York: Wiley.
- Schneider, S.J., Cantor, D., Arieira, C., Malakhoff, L., Segel, P., Nguyen, L., and Guarino, J. (2002) *An experiment comparing computer-assisted and paper modes of data collection for the short form in Census 2000*. Paper presented to the American Association for Public Opinion Research, St. Pete Beach, FL, May 17.
- Schumacher, R.M., Hardzinski, M.L., & Schwartz, A.L. (1995) Increasing the usability of interactive voice response systems: Research and guidelines for phone-based interfaces. *Human Factors*, 37, 251-264.
- Tourangeau, R. and T. Smith (1998) Collecting sensitive information with different modes of data collection. In: Couper, M. Baker, R., Bethlehem, J., Clark, C., Martin, J. Nichols, W., and O'Reilly, J. (eds.) *Computer assisted survey information collection*. New York: Wiley, pp. 431-454
- Tourangeau, R., Steigler, D., and Wilson D. (2002) Self administered questions by telephone: Evaluating interactive voice response. *Public Opinion Quarterly*, 66, 265-278.
- Turner, C. F., Ku, L., Rogers, S. M., Lindberg, L. D., Pleck, J. H., and Sonenstein, F. L. (1998) Adolescent sexual behavior, drug use, and violence: increased reporting with computer survey technology. *Science*, 280, 867-873.
- Weinschenk, S. and Barker, D. (2000) *Designing Effective Speech Interfaces*. New York: Wiley.