# Web-Based Surveys for Corporate Information Gathering: A Bias-Reducing Design Framework

—JAKE BURKEY
AND WILLIAM L. KUECHLER

***Abstract*—*The cost effectiveness of internet-based communications in the ever more fully networked business environment continues to drive the use of web surveys for corporate information gathering. However, simply applying traditional survey techniques to the web can result in significant shortcomings in the data so gathered. Recent research has been directed at these issues, within the context of web surveys as a general research tool. We discuss the application of that research to the narrower organizational context. This article synthesizes from the literature on web-based surveying and from the authors' own experiences. It suggests a design framework for managers and communications professionals interested in increasing the statistical validity of web surveys deployed in an organizational context. The results of a recent organizational web survey developed and administered within the framework guidelines support the efficacy of the framework.*

***Index Terms*—*Cross-browser compatibility, nonresponse error, research bias, research methodology, survey methodology, web research, web survey.*

Manuscript received August 5, 2002; revised December 1, 2002.

J. Burkey is with the University Center for Economic Development, Applied Economics and Statistics, University of Nevada, Reno, NV 89557 USA (email: jburkey@unr.edu).
W. L. Kuechler is with the Department of Accounting and Computer Information Systems, University of Nevada, Reno, NV 89557 USA (email: kuechler@unr.edu).

IEEE DOI 10.1109/TPC.2003.813167

Web surveys are an information-gathering technique that is used increasingly both within and across organizations and groups. Although some larger corporations have been using computer-based surveys for many years, greater compatibility across web browsers and free or inexpensive software have made the technology affordable and attractive for small and mid-sized organizations as well [1].

Cost is prominent among the factors that motivate an organization to explore the use of web surveys. Dillman notes that the two major advances in survey methodology in the twentieth century were the introduction of random sampling in the 1940s and interviewing by telephone in the 1970s, and that the development and implementation of both were driven by the desire to minimize cost [2]. Even the most basic web-survey design allows for the presentation of an online survey document as if it were a self-administered mail survey on a printed page. However, with a modest design effort beyond the basic information presentation, a web-based survey offers the potential to interact with the respondent as would a surveyor in a face-to-face interview, providing assistance and checking for completeness of answers, without the same potential for bias that occurs in personal interviews. More importantly for employee surveys, the web survey can incorporate the strengths of personal interviews while maintaining respondent anonymity. Paper, postage, printing, and data entry costs may be reduced or eliminated, as there is the close correlation between sample size and survey costs. Once the electronic data-collection system is developed and implemented, the marginal cost (i.e., the cost of surveying each additional person) is extremely small [2]. Since many organizations already have networked systems, existing server capacity may be used for the survey, and issues such as access to and familiarity with the computer technology required to complete the survey are of less significance. Cross-platform compatibility issues are also reduced, though not eliminated, as even organizations that standardize on a single operating system may have some exceptions to that standard [3].

Methodological benefits associated with web surveys are another appealing factor and include allowing the choice set presented to each respondent to be defined uniquely in order to accommodate the use of certain statistical methods of analysis. For example, researchers in the Department of Applied Economics and Statistics at the University of Nevada, Reno (UNR), have used web surveys in nonmarket valuation studies that incorporate contingent behavior questions requiring the presentation of unique dollar amounts to each respondent.

Subsequent to a comprehensive review of the literature on the techniques of web surveys, it has become apparent that most papers have focused on the use of web surveys as general research tools rather than as information-gathering tools for organizational decision making. The strengths of web surveys in an organizational context, as compared to surveys of the general population, make them particularly appealing and appropriate for data collection. The authors' interest in the techniques by which the validity and accuracy of web-based survey results could be made to compete with traditional survey techniques in an organizational context began with a combined contingent valuation and revealed preference analysis of campus parking, based on a web survey of the faculty, staff, and students at UNR. Techniques learned and refined in the development and administration of subsequent web surveys have been incorporated with the findings of other researchers to form the design framework presented in this paper.

## OVERVIEW OF THE DESIGN FRAMEWORK

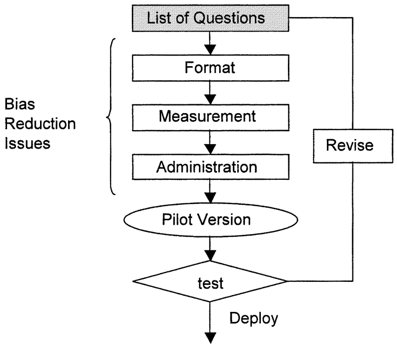
Church and Waclawski provide detailed instructions on the development and administration of organizational surveys, but they include little coverage of web surveys, dismissing their appeal as "sexiness" and citing security and anonymity issues as obstacles [1]. Morrel-Samuels provides a general set of guidelines for the development of organizational surveys, mentioning the web only as a possible delivery mode [4]. He identifies development issues within five areas. (1) CONTENT encompasses the processes by which the survey questions are selected. (2) FORMAT relates to the manner in which the questions are presented. (3) LANGUAGE covers the wording of the questions. (4) MEASUREMENT addresses the design of response scales. (5) ADMINISTRATION touches on issues related to the delivery mode of the survey. Content and language are not affected by survey mode and will be covered only briefly in this paper. Format, measurement, and administration will be discussed to the degree to which they present unique issues for web surveys.

Fig. 1 illustrates the application of Morrel-Samuels framework as a functional taxonomy for the classification of web-survey design guidelines and the incorporation of research results from the current literature [4]. This taxonomy is applied within a standard iterative software design methodology consisting of development phases that are revisited as a prototype survey is refined [5]. The generation of a list of survey questions, screened for content and language, and formatted in accordance with general guidelines, comprises a starting point for our framework and has been absorbed in the "gray box" labeled "List of Questions" in Fig. 1.

The structure of the paper from this point follows the format of the framework. Potential sources of bias, which must be attended to throughout the design process, are considered first. The issues unique to web-based surveying, those which focus on format, measurement of data, and survey administration, are then addressed. A discussion of the prototype survey under the framework includes the testing of the layout and content of the questionnaire. The hardware and software components of the web application that delivers it to the respondent and that receives and processes the response data are also tested, and the results are reported. A concluding section discusses general issues and summarizes the key design issues in Table I.

To provide empirical evidence for analysis of the framework and its methodology, we provide results from our most recent organizational web survey. This survey (the framework

Fig. 1. Web survey development framework.



evaluation survey), conducted for a small telecommunications firm (ca. 100 employees), was designed and administered in full accordance with the framework. The multimodal delivery of the survey additionally provides for the ability to compare paper and web survey modes in a controlled manner.

## Bias in Web Surveys

Couper has identified several types of web surveys within the general categories of nonprobability and probability-based surveys [6]. Nonprobability surveys do not select a representative sample and, thus, cannot produce conclusions that may be generalized to a population. Nonprobability survey types include polls as entertainment (e.g., news opinion polls), unrestricted self-selected surveys, and volunteer opt-in panels [6]. Probability surveys, conversely, are designed to obtain a representative sample of the population. Probability-based methods include intercept surveys, list-based samples, web option in mixed-mode surveys, prerecruited panels of internet users, and probability samples of the full population [6]. Organizational surveys that will be used in guiding policy must be probability-based to be truly representative, but other survey types may be appropriate for an organizational survey depending upon the need.

In probability-based surveys, findings from a sample of a population may be generalized to the entire population using statistical inference, allowing us to say with measurable certainty that people do something, or that they will do something, or to calculate the amount of the thing they will do. Bias damages the accuracy and precision of that inference, and may lead to incorrect conclusions. Groves [7] was the first to describe and categorize bias relevant to surveys, and Dillman has employed the same structure to describe bias [2]. Couper, in a detailed discussion of the types of bias important for web surveys, modifies Groves' structure by combining two very similar categories of bias to yield

TABLE I
Summary of Web Survey Bias Introduction Issues With Design and Implementation Approaches

| Web development stage and potential problem | Design level solution | Implementation Possibility |
|---|---|---|
| ***Format*** | | |
| **Survey nonresponse** | Maximize ease of entry and continuation | ▪ Scrolling between questions on a single page<br>▪ Initial questions easy and or interesting<br>▪ Motivational, explanatory information on page(s) |
| *Skipped questions and/or skewed distribution of results* | Provide information for question topics as well as the form controls | ▪ Text instructions<br>▪ DHTML boxes<br>▪ JavaScript pop-ups (least desirable; see *Format*) |
| ***Measurement*** | | |
| *Skipped questions and/or skewed distribution of results* | Enforce consistency of display | ▪ Cascading Style Sheets (CSS)<br>▪ Document Object Model standard for DHTML |
| | Allow for adaptation to lowest common denominator | ▪ Nested tables for maintaining spatial relationships between display elements<br>▪ Client-side script to determine browser attributes and correct programmatically |
| ***Administration*** | | |
| *Survey nonresponse and/or skewed distribution of results* | Test system under load | Check with in-house IT group or 3rd party host. Software is readily available to perform simulated stress testing. If heavy, variable loads are anticipated, live testing is preferred. |
| | Decrease time for pages to load | ▪ Eliminate unnecessary graphics.<br>▪ Use standard HTML controls whenever possible.<br>▪ Validation and programmatic prompting on the client using scripting languages as opposed to server side programming. |
| | Increase perception of anonymity | ▪ 3rd party hosting of site<br>▪ Common, shared entry terminals (other than the respondents personal work terminal)<br>▪ Allow access from World Wide Web. |
| *Missing or corrupt data* | Prepare form results for database entry and identify skipped questions | Client or server side scripts can place program-recognizable entries in skipped data fields |
| *Sample selection bias; sample not representative of population* | Admit only respondents with a pre-assigned unique access code | Generate unique access codes and distribute one to each participant. Each access code is detected prior to form issuance and can be used only once. May raise anonymity issues (see text). |
| | Obtain census | Offer all members of organization the opportunity to participate in the survey |
| | Obtain stratified sample | In large organization, proportion of participants across departments or employment classes is the same as in the entire organization |

three general types: coverage and sampling error, nonresponse error, and measurement error [6].

Morrel-Samuels [4] presents guidelines to prevent or reduce the bias defined by Couper [6] but does not describe how that bias arises within the developmental framework. Understanding the types of survey bias allows for the improvement of the survey process through the development of a set of guidelines tailored to the needs of the individual organization. The discussion below focuses on the three types of survey bias as they affect web surveys, specifically within an organizational context.

**Coverage and Sampling Error** Coverage and sampling error is associated mainly with the administration phase of the survey process. It occurs because of a mismatch between the target population and the sample population, a significant issue for web surveys of the general public. For example, results from numerous surveys show that about 30–40% of households use the internet and that this group is not representative of the general population [6]. In a web survey of the general population, therefore, there is no possibility of drawing a sample in which nearly every adult in the population has a known nonzero chance of being selected for participation in the survey because the percentage of households with internet access in the U.S. population is still significantly less than 100% [2]. To the extent that employees have access to the web within the workplace, lack of access should not represent a significant source of bias for most organizational web surveys. Sampling within a larger organization may be stratified in order to obtain a representative sample across employment categories, or census data may be obtained by surveying every member of an organization, eliminating most or all coverage and sampling error. Due to the relatively small organization size in this study, we chose to administer the framework evaluation survey as a census, eliminating the need for the selection of a probability-based sample.

**Nonresponse Error** Two types of nonresponse behavior have been delineated in the literature: UNIT NONRESPONSE, in which the respondent fails to participate in a survey, and ITEM NONRESPONSE, in which a respondent fails to answer one or more questions in the survey. As discussed below, NONRESPONSE ERROR will refer to the sample selection bias introduced by unit nonresponse. Item nonresponse contributes to incomplete data sets but does not directly introduce bias.

Nonresponse error may result from decisions made at any stage in the development process. Anything that compels a respondent not to complete the survey, such as offensive wording, confusing format, lack of anonymity, or poor server performance, will contribute to nonresponse error. Nonresponse error in web surveys of the general population has been shown to occur at very high rates relative to other forms of surveys [6], and some studies have shown that, given a choice between a web survey and a mail survey, most respondents choose the mail survey [8], [9]. In a more detailed look at the issue, a recent study by the Council for Marketing and Opinion Research (CMOR) in which respondents stated preferences for survey methods found that among respondents contacted via the internet, 78.9% chose “internet” as their first-choice method, and among respondents contacted by telephone, a 39.9% majority selected “mail” as their first-choice method [10], [11]. This seems to imply that individuals who are currently using the internet have a strong preference for it, a finding that bodes well for organizational web surveys. It also suggests that a carefully designed mixed-mode survey, using a combination of the web and mail, might achieve much higher response rates than would either mode alone. Having a choice of response mode may in itself be a motivator toward participation, and the CMOR study points to mail and the web as the two most popular response modes [2].

Nonresponse error may be significant for reasons other than survey mode alone, however, and in an organizational setting, the respondents’ perception of anonymity is probably the factor most highly correlated with nonresponse [1], [4]. In an attempt to maximize the perception of anonymity, the framework evaluation survey was implemented as an anonymous survey, with access codes distributed in unmarked sealed envelopes. The survey was administered remotely by the UNR Center for Economic Development, and respondents were given the choice of responding by mail or by web.

Other important factors inducing nonresponse are tied to the design of the website and survey instrument, including ease of use and time required to completion [12], [13]. One of the primary goals for our framework evaluation web questionnaire was simplicity and ease of use. The pilot test results confirmed that we had been largely successful in achieving that goal. The target for the mean time to completion was 20 minutes or less, as recommended by Morrel-Samuels [4]. In addition, our overall response rate of 79% compares favorably with other published figures [12]–[15]. Completed surveys were fairly evenly distributed between the mail and web response modes, and a comparison of the distributions of responses for the two modes, using tests for central tendency and dispersion, showed no statistical difference. Results from completed surveys suggest that there is no difference between the data collected via mail or web.

**Measurement Error** Finally, where content and language impacts have been addressed, measurement error in the web survey will stem from format and measurement issues, such as response scale design or the correct display of a response scale in different size browser windows or different resolution monitor displays. Many researchers have documented the effect that different interfaces can have on survey data. For instance, designers of web-based surveys must consider the order or position choices [2], [16] and instructions for conditional questions [17]. For these reasons, format and measurement are the most important types of bias to consider during the design and construction of the web survey instrument, regardless of the target population. Part of the appeal of web surveys is the capacity for interactive administration, absent from the biases induced by human surveyors, but research by Couper [18] shows that people interact with webpages in a manner similar to interactions with other humans and that design factors are important in reducing such bias. Techniques for minimizing the bias that can result from poorly designed interfaces are described in the next section.

**Format** One of the most common and most easily implemented recommendations is that the site should be conservatively formatted, avoiding the use of multiple colors and shapes [19], [20]. We implemented style formatting across all the pages on our framework evaluation survey using a global cascading style sheets (CSS) document, a text file containing formatting instructions directly incorporated into all HTML tags of a specified type or applied through the class attribute of individual tags. Although the implementation of CSS formatting commands is more uniform across the two major browsers (Internet Explorer and Netscape Navigator) than in the past, it is not yet entirely the same on both, and a survey designed to be cross-browser compatible should be checked for display on both browsers. Since the code for the survey will not require much modification once built, if cross-browser compatibility is an issue and CSS formatting is not being uniformly rendered, it is worth considering the use (at least in part) of older "deprecated" HTML formatting tags, still supported by both browsers, and often more reliably implemented. Dillman [2] [19] and others [4], [12], [16], [17] also provide guidelines for page layout. For example, questions should be numbered using a bold font to the left of the question. Other recommendations address the placement of answer boxes or radio buttons and the grouping of questions on a page. Most of the guidelines are not specific to web surveys but are appropriate for any published questionnaire, whether printed or displayed on a screen, and should be implemented on a webpage as on a printed page. Following those recommendations, we designed the framework evaluation questionnaire with black question text against a plain white background anchored at the left by large, bold, brightly-colored numbers, with ellipses joining the text to the answer fields at right, and with no double-banking of questions. Selection of HTML controls was determined by question type, using radio buttons for Yes/No type questions, and text boxes for most other questions (see the Measurement section following for more on this). Responding to a focus group request, we also incorporated text area controls for a set of questions that allowed multiple responses to a single question. This required separate scripting commands across the two major browsers and the Windows and Macintosh operating systems due to differences in how new lines in the text area are assigned. Our mail questionnaire was printed as an exact duplicate of the framework evaluation web questionnaire, with the exception that the text hidden by the DHTML skip controls was visible on the printed survey.

Another very common recommendation is that an attempt should be made to standardize display across different screen configurations, operating systems, browsers, and partial screen displays [2], [21]. Even with an organization-wide uniform browser and operating system, without a resizable screen display, answer fields may be off the visible part of the screen or may not line up with the text, depending on screen resolution and browser window size. In creating a questionnaire to adapt to browser window resizing while maintaining proper alignment of the form elements, we adopted a passive approach relying on the use of nested tables to simplify coding and assure cross-browser compatibility. Formatting instructions that set the tables to occupy a certain percentage of the screen width and have specified margins are contained within the CSS document. For a very complex questionnaire requiring many levels of nesting, it may be worthwhile to use CSS for the sizing and formatting of all elements. The Wired.com news site (http://www.wired.com/news/explanation.html) recently made the transition to a page design formatted entirely through CSS, which appears to handle browser window resizing nearly as well as nested tables.

Another consideration is whether the questionnaire will appear on a single page or across multiple pages. Dillman [2] recommends that the questionnaire be structured so that it scrolls from question to question, preferably on a single page. Morrel-Samuels [4] makes a similar recommendation. Scrolling speeds up the form completion process as it requires less interaction with the server and it allows the respondent to assess progress by noting the position of the scroll bar. Our framework evaluation questionnaire was designed accordingly, with questions presented linearly down a single page; this practice may minimize respondent quitting of web surveys with only a few questions left [2], [13].

The first visible page of the web survey should be motivational, with a brief explanation of the reason for the survey, encouraging the viewer to respond and emphasizing the ease of responding [2]. It is very easy to exit any webpage, so the first screen must be designed to help people get to the content of the survey as quickly and easily as possible. Likewise, the first question should be interesting and easily answered. As with mail surveys, demographic or background questions such as education, age, and income should not appear at the beginning of a survey [4], [22]. In a study by Frick et al., a web questionnaire with sociodemographic questions positioned at the beginning had a drop-out rate of 17.5%, while the version with those questions positioned at the end had a drop-out rate of 10.3% [23].

The time required to complete the questionnaire should be less than 20 min and should be checked from a web terminal rather than a local drive (see also the section on pilot testing) [4]. Additional download time for graphics or calls to server-side scripts will significantly increase the survey completion time. In the analysis of a multipage questionnaire format, Crawford found that a small graphic used to indicate progress significantly increased average survey completion time and was negatively correlated with survey completion [13]. Our framework evaluation questionnaire contained only one very small image, which loaded only once upon initial page load since the survey was not a multipage design. Finally, though computer-administered questionnaires have been found to take longer than pen-and-paper, Fuchs et al. found that most of the time difference can be attributed to differences of design between the two methods, rather than the technology itself, implying that careful design can substantially reduce the additional time burden imposed by web surveys [24], [25]. The results of our framework evaluation survey appear to support that conclusion, with a mean time of 13.9 minutes for the web questionnaire versus 34.3 minutes for the mail questionnaire. The time to completion of our web questionnaire was recorded using a server-side script, while the time to completion for the mail questionnaire was self-reported. As expected, the variance for the mail questionnaire is greater than that for our web questionnaire.

Instructions regarding necessary respondent actions should be provided, with additional instructions at each point where needed [2]. Our framework evaluation questionnaire incorporated this recommendation in three parts. First, a short paragraph of general instructions was given at the top of the questionnaire. Second, the questions themselves were then carefully worded to provide additional guidance. Third, pop-up help windows were created for each question to provide additional insight into the specific question being asked. The first two instructional components were incorporated into the mail questionnaire using an identical format, but the text from the help windows had to be printed to a separate sheet of paper. The mail questionnaires received exhibited a significantly higher rate of errors and incomplete answers than did the web results, even though the web questionnaire validation script did not force answers to any of the questions. This suggests that the pop-up help windows make information more intuitively and readily available to respondents; this is an area into which we will be conducting further research.

Help windows are an ideal means of leveraging the interactive capabilities of web questionnaires. There are several ways to create such pop-up windows, the most common being JavaScript pop-ups and dynamic HTML (DHTML) boxes. We used DHTML in our implementation. It is faster to appear, as the text loads with the page and is kept hidden until requested by the client. JavaScript pop-ups force the browser to open a new window, a slower process. Also, clients may disable the browser’s implementation of such scripts, in response to annoying pop-up advertisements, rendering a script-based design unusable. The issue of cross-browser compatibility was significant for our web surveys, as we also used DHTML to eliminate the need for skips and skip instructions through dynamically adaptive forms that would display certain questions conditional on responses to others. Alternatively, in a questionnaire in which respondents are intended to skip specific questions and DHTML will not be used to implement the skips, skip instructions should be provided in a manner that encourages selection of the answer to the skip question prior to clicking on a link to accomplish the skip [2]. In addition, the type and formatting of questions associated with skips on self-administered questionnaires are significant in influencing skip-pattern compliance [26], [27].

DHTML was also used as part of the framework evaluation questionnaire form validation script to highlight questions left unanswered. The first browsers from Microsoft and Netscape to implement DHTML (version 4) did so in very different ways, and the use of it in those browsers necessitated the creation of two sets of source code. That has changed, however, as more recent versions of both browsers now implement the international standard of the Document Object Model (DOM;

http://www.w3.org/DOM/) to a significant degree, allowing the use of a single script to produce DHTML effects that are almost identical on both browsers. With that in mind, and assuming that most respondents to our latest survey would be using at least version 5 browsers, our strategy for the use of DHTML involved writing it using DOM methods to address the page elements. Upon receiving a request for the survey page, a server-side script would determine the browser model and either load the requested page for DOM-compliant browsers, or load a custom error page for noncompliant browsers, containing a message explaining browser requirements for access to the site. An alternative, employed in some of our earlier web surveys, directed noncompliant browsers to an HTML-only page with written skip instructions on it. The use of this strategy invites potential measurement error due to the differences between the two questionnaires, however, and should be avoided, if possible.

## MEASUREMENT

All HTML form controls can be used for collecting answers to questions, but each type has its strengths and limitations. Text boxes allow for open-ended responses, but may require a validation script to check for the existence of a response, or a parsing script to put the response into a usable form. In general, controls such as drop-down menus, radio buttons, and check boxes allow for simpler data handling in that responses are selected from a discrete choice set and validation is easier. There are important considerations in using such controls, however. Drop-down menus should be used sparingly and are not appropriate for binary questions, such as gender, where a set of radio buttons requires less effort to answer [2]. Another strength of radio buttons is that they may be visually arrayed in a set, allowing for the construction of response scales such as those described in Dillman [2] and Morrel-Samuels [4], and discussed in greater detail in Schwarz [28]. A limitation of both types of controls is that the first answer of a drop-down menu or radio button choice set should not appear by default as the selected choice. For drop-down menus, it is easy to disallow the first choice as an answer, but for radio buttons, it is less straightforward. Ideally, the page should load with no radio buttons checked, but if a validation script is needed to check for the existence of a response, the radio button set cannot be examined by control name or I.D. if none of the radio buttons is checked. Our validation script works around this issue by checking the elements array of the form object, sequentially testing the elements whose type is "radio."

**Custom Controls** "Standard" HTML form controls have the advantage of being similarly displayed in all common browsers. For surveys where budgets permit custom programming, however, custom controls can be developed in Java and embedded in HTML pages as applets (see Francis et al. [29] for a development example). For corporate surveys where the browser and server environment are closely controlled and use Microsoft software, custom controls, and the .NET framework, which includes many more controls than standard HTML, can be used. Additionally, HTML controls can be used in creative ways. For example, Likert response scales can be implemented as multiple-linked radio buttons, or as default responses in text entry boxes where users can place an "X" by moving the cursor to the appropriate spot on the form just as they would with a pencil in a paper survey.

Custom controls have the advantage of a more sophisticated appearance and the potential for ease of use. For example, a custom control with a "slider" that can be moved with the mouse giving constant visual display of a numeric entry corresponding to slider position may seem to be a good solution to entry of numeric or Likert scale data. Users may not be familiar with the use of very sophisticated controls, however, and user response to these controls is a subject of ongoing investigation. Additionally, all custom controls require more sophisticated programming to gather the data and usually take more time to load into the users' browser, thereby increasing the possibility of nonresponse bias. Unless there is an exceptional reason to do otherwise, we recommend the use of standard HTML controls and a design that permits responses as analogous to paper form responses as possible.

## ADMINISTRATION

As shown in Fig. 2, a web survey is an example of a client-server hardware and software system. Separate programs are required for the browser client and for the server. The client software, typically the HTML and JavaScript questionnaire form, is downloaded from the server. Thereafter, it runs solely on the client system until survey results are submitted (sent back to) the server. As discussed above, we made extensive use of client-side scripting in the interest of reducing response time to client-side events. Server software is responsible for extracting data from the form, storing them into a database and optionally performing data analysis. Server side software also frequently performs some validation of form input, as discussed in more detail below.

Client-side development, server-side development, or both can be outsourced, resulting in three general options for design and deployment of organizational web surveys: contracting the job out to a polling firm, using one of the many web-based polling sites, or developing and implementing the survey in-house. The option chosen will depend on the nature of the survey and the size and type of the organization, among other things. However, an understanding of the survey development framework and the web-survey development process and issues set out in this paper are required whether an organization is developing and administering the survey in-house or using a third party for some or all of the process. Even in a third party situation, the ultimate responsibility for the accuracy of the results rests with the contracting organization.

Surveys of large organizations are frequently conducted by a third-party professional polling firm. Crawford and Morrel-Samuels are both employed by such firms, and their contributions to the literature reflect the practical experience gained as a result [13], [4]. One of the most important advantages of this approach is that it may increase respondents' perception of the anonymity of responses.

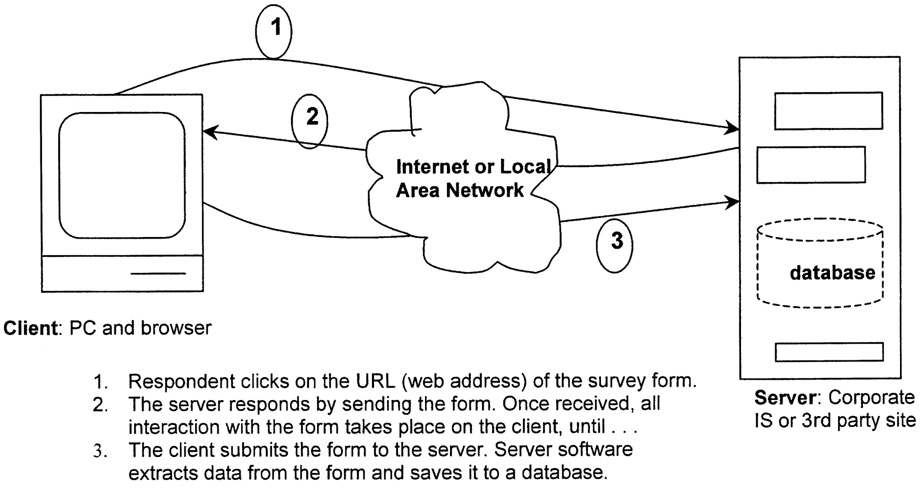
Many organizations are interested in administering their own web surveys, however, and must either purchase survey software or write the code necessary for implementation of the survey. Either option will require one or more employees dedicated to the task. While many organizations have full- or part-time website administrators for their information portals, the skill set and software techniques required for web surveying are different and more sophisticated than those required for maintenance of webpage content. Any organization conducting sales or gathering client information over the web, however, will probably have the requisite skills for survey administration in-house.

**Validation** One of the strengths of web surveys is the ability to enforce validation, which helps ensure that the data collected are clean and complete. Although Dillman strongly recommends not forcing respondents to provide an answer to each question in order to submit the survey [2], [19], little research has been done to quantify the impact that this may have on response rate. Dillman suggests that although this capability inherent in web surveys is often presented as an advantage, it is likely to result in premature terminations because by forcing an answer, the surveyor introduces a stimulus which is not present in other forms of surveys and which has not been evaluated for bias. That is supported by the findings of a study by Bosnjak and Tuten, which examined the types and amount of nonresponse in a web-based survey [30]. In that study, 36% of participants who completed the survey left one or more individual items unanswered, and only 25% of respondents provided answers to every question. It is not clear, however, how much of the item nonresponse may have been due to the content of the question as opposed to the visual presentation of the question, which Dillman et al. have shown to be a significant contributor to item nonresponse [31].

We have found that rigorous validation, regardless of whether it is used to prompt the respondent for further information, is of substantial value in preventing errors as the data are being written to the database. In our experience, the greatest source of lost data and premature terminations has been unforeseen database errors related to data types and data length. Validation can be useful also in preventing server errors when writing to the database by making sure that a "no data" value is written to fields left empty by the respondent. This practice will help in the analysis phase as well by distinguishing between answers of zero and nonanswers.

Validation scripts may take any form imaginable by the designer,

Fig. 2. Client server architecture for web-based surveys.


1
2
3
Internet or Local Area Network
database
Client: PC and browser
Server: Corporate IS or 3rd party site

1. Respondent clicks on the URL (web address) of the survey form.
2. The server responds by sending the form. Once received, all interaction with the form takes place on the client, until . . .
3. The client submits the form to the server. Server software extracts data from the form and saves it to a database.

and instead of forcing a response, may be written to simply prompt for a response, allowing the submission to proceed with no response after the prompt. This method produced the most complete data in a study of the issue by DeRouvray and Couper, and is the method we have employed in all of our recent web surveys [32]. Allowing for no response is most important with regard to demographic questions such as income. Many individuals will refuse to provide an amount for income and may abandon the survey when presented with that question.

In order to prevent multiple responses or responses from individuals outside the survey sample, access to the survey must be restricted. The use of an access code is the best way to restrict access to the survey to that group of respondents chosen by a probability sampling method or census [2], [13], [33]. That was accomplished in our framework evaluation survey by creating two tables in the database, one for user information and one for survey data, using the unique access code as the primary key for both. Upon submission of an access code from the form on the login page, a server-side script would check for the existence of the code among the records in the database. If the code could be found, the script would then determine the submission status of the record. One of the features new to the framework evaluation survey, requested by focus group members during the design phase, was functionality that allowed the partial completion of the survey so that it could be completed in more than one session. The database would store the partial results, displaying them to the respondent upon return to the survey. Upon final completion of the survey, a status field in the record would be updated to reflect that. If the access code was detected and the data record was not completed, the script would allow the questionnaire to load. Otherwise, it would load a custom error page with a message related to the specific error. Any attempt to load the questionnaire directly without going through the login page would likewise result in a specific custom error message.

In addition to recording submission status, the user table retained data on the number of times a user accessed the survey, as well as the time and duration of each session.

**Development and Administration Cases** Our experience suggests that survey software development tools are unlikely to make the task any easier. They are almost as difficult to master as building a website using code and are much less flexible. However, development tools have been used in a number of cases in the literature, and we provide an overview of those experiences below.

The choice of operating system on which to develop the survey and administer it is usually beyond the control of those most concerned with the survey and its results. However, a variety of tools are available for both UNIX/ LINUX and Microsoft platforms. Since Macintosh systems now utilize a UNIX operating system, these tools are applicable for a large majority of the survey platforms currently in use.

Dennis and Gambhir designed a database-driven web survey application called iQ&A, built on the freely available AOLserver web server software and the free version of the Sybase database software [34]. Couper et al. used a program from Surveycraft called ScyWeb [12]. Other researchers cited have built web survey sites from a variety of free or inexpensive software readily available to any organization. Questionnaire forms are usually written in HTML, and client-side validation scripts or DHTML effects are nearly always written in JavaScript, which is the native scripting language for both of the major browsers. Server-side scripts are needed to transfer the data collected by the form into a database and may perform other functions such as validation of access codes.

Heerwegh and Loosveldt used a PHP script, which is a popular choice for UNIX platforms [33]. PHP is commonly used in conjunction with the MySQL database, both of which are free, although Heerwegh and Loosveldt did not identify the database used in their study. They did mention, however, that one of the functions of the PHP script was the validation of a four-digit access code. Francis et al. ran the free Apache web server on a free Linux platform and collected data via Java applets, although they described the coding process as difficult [29].

In the Microsoft environment, Crawford et al. used Cold Fusion with a Microsoft Access database, contained in the Microsoft Office suite [13]. In addition, Microsoft's Active Server Pages (ASP) and the newer ASP.NET are available for free, and work well when paired with an Access database on a Microsoft server. The Windows 2000 operating system is sold with a limited version of Microsoft Internet Information Server (IIS) installed; this is adequate for the needs of smaller organizations. We have used ASP with an Access2000 database for our more recent surveys, running on a full version of IIS already in place as the server for the department network.

## PILOT TESTING

Pilot testing, indicated by the decision diamond in Fig. 1, examines the readiness for deployment of not only the survey instrument itself (the HTML and JavaScript survey form downloaded to the client) but the data collection software on the server and the entire survey administration process. The number and rigor of tests applied to the pilot survey will vary depending on whether or not it has been designed as a probability-based survey. All web surveys should be tested for issues related to the downloading and display of the questionnaire within the client browser as well as the processing of data received by the server.

**Performance and Format Testing** A key performance issue is the testing of the complete system under simulated load. The server may slow or crash when the survey is initiated, introducing measurement error into the responses received and nonresponse error from the reduced number of completed questionnaires [13]. In a recent organizational web survey of the staff at Saint Mary's Regional Medical Center, Reno, many employees were unable to access or complete the survey due to the inability of the server to handle the load. Regardless of whether an outside firm is being hired to administer the survey, the questionnaire should be tested for performance in-house under simulated load.

In addition to testing the server's ability to deliver web content under load, the server-side processing of data must be checked to confirm that the data received from the web form are being written to the database correctly and completely. During the administration of our parking survey, we discovered that a small number of fields in the database were not being populated with data due to a script error. The error had been missed in an earlier check of the database because each individual field in the form had not been tested. Even after performing a thorough test of all form fields, we have experienced occasional unexpected database errors, such as those resulting from a longer than expected number or string input. As mentioned previously, a rigorous form validation script is critical for the prevention of database errors.

The layout of the webpages should appear as expected, and all hyperlinks and scripts should be checked. It is best to design the survey to be viewed in multiple browser versions, as different browsers may be in use regardless of whether the organization has a standardized software platform [3]. Some variation in display across browsers may be acceptable, however, depending on the type of survey and the nature of the variation.

As a final exhortation to rigorous pilot testing, we note that for most surveys there is no "shakedown" period. Each survey is unique and time-bound, and once initiated, every error is either a missed response, a source of bias, or both.

**Statistical Validity Testing** Pilot data from probability-based surveys must also be examined within a statistical context. Foremost among a few simple but informative tests mentioned by Morrel-Samuels is checking for the distribution of data along a normal bell-shaped curve [4]. Data that exhibit a skewed distribution will be less useful in an analysis and are probably due to a poorly worded question or the perception that the survey is less than completely anonymous. Lengthy surveys will also produce automatic and overly positive results, visible as a skewed distribution of data [4]. Nonresponse, both as item nonresponse and survey nonresponse, is also a primary concern. Item nonresponse may reveal issues related to questionnaire content, format or length, or survey anonymity. Survey nonresponse within an organization is most likely related to the perceived degree of anonymity, but may also be related to server and browser function.

If time and resources permit, the best way to test the pilot web survey for statistical bias is to administer it in parallel with other modes, such as phone or paper-based questionnaires, and then compare the shape and position of the resultant distributions across modes. In a 2001 pilot test by Duke Energy, in which email and paper-based questionnaires were administered in parallel, data from the email mode had a higher mean, narrower range, and more unanswered questions due the perceived lack of anonymity of that mode [4].

The pilot test may be the most important step in web survey development, and attention to detail at this point can ensure that the survey produces enough usable data to enable a meaningful analysis. Following the iterative methodology in the design framework will ensure that errors detected in pilot testing will be corrected **and** that the effects of the corrections will be followed through all design phases, including another pilot test. As with any system, seemingly simple modifications can ripple through the design to create unanticipated consequences in other areas.

## SUMMARY

The web as a new survey mode holds promise in several areas, including cost-efficiency, reduction of certain forms of bias, and versatility. The challenges facing the broad implementation of web surveys can be seen as belonging to three general categories, including sample selection, cost to the respondent, and bias.

Table I lists in summary form, by framework category, some of the more important bias issues for web surveys and general design level approaches for minimizing the effect of these issues. The high level approaches are followed by suggestions for implementing the design level solutions. However, we emphasize that these are suggestions only; actual implementation possibilities are both constrained by deployment platform and are highly variable depending on the knowledge and skill of the implementation team and the tools available to them.

At this point in time, the significance of sample selection issues is a function of survey scope and presents the greatest obstacle to surveys attempting to generalize web surveys to the full population. Survey scope presents far less of a problem in an organizational environment. As organizations become ever more fully networked, and increasingly sophisticated web-survey development tools (and in-house skills) become widely available, survey costs continue to drop, and web surveys become increasingly more applicable. Bias issues unique to web surveys must be addressed at each stage of the development process. Through the use of the design framework presented in this paper, careful attention can be directed to the critical elements of a web survey, increasing the value of the data obtained.

Throughout the paper we have indicated how our evaluation survey, incorporating all the design elements from our framework, has performed relative to published web survey measures. Although surveys are necessarily unique and comparisons must be made cautiously, the evaluation survey has provided valid and significant results that compare very favorably with both published survey performance analyzes and our own survey experience prior to development of the framework.

## ACKNOWLEDGMENT

The authors would like to thank Professor Thomas R. Harris, UNR Center for Economic Development, for allowing us the control of survey design and the use of data from the organizational economic survey employed for framework evaluation.

## REFERENCES

[1] A. H. Church and J. Waclawski, *Designing and Using Organizational Surveys: A Seven-Step Process*. San Francisco, CA: Jossey-Bass, 2001.

[2] D. A. Dillman, *Mail and Internet Surveys: The Tailored Design Method*. New York: Wiley, 2000.

[3] D. Zatz, "Create effective email surveys," *HR Magazine*, vol. 45, no. 1, pp. 97–103, 2000.

[4] P. Morrel-Samuels, "Getting the truth into workplace surveys," *Harvard Bus. Rev.*, vol. 80, no. 2, pp. 111–118, 2002.

[5] R. S. Pressman, *Software Engineering: A Practitioner's Approach*. New York: McGraw-Hill, 1997.

[6] M. P. Couper, "Web surveys: A review of issues and approaches," *Public Opinion Quart.*, vol. 64, pp. 464–494, 2000.

[7] R. M. Groves, "Research on survey data quality," *Public Opinion Quart.*, vol. 51, no. 2, pp. S156–S172, 1987.

[8] S. D. Crawford, The Web Survey in a Mixed Mode Data Collection. Ann Arbor, MI: Univ. of Michigan, 1999.

[9] H. Tedesco, R. L. Zuckerberg, and E. Nichols, "Designing surveys for the next millennium: Web-based questionnaire design issues," in *Proc. 3rd ASC Int. Conf.*, Sept. 1999, pp. 103–112.

[10] J. M. Sheppard. (2002, March 6) 2001 Respondent cooperation and industry image study: Privacy and survey research. The Council for Marketing Opinion and Research. [Online]. Available: http://www.cmor.org/cmorrestudtoc.htm.

[11] S. Jarvis, "CMOR finds survey refusal rate still rising," *Marketing News*, vol. 36, no. 3, p. 10, 2002.

[12] M. P. Couper, M. W. Traugott, and M. J. Lamias, "Web survey design and administration," *Public Opinion Quart.*, vol. 65, pp. 230–253, 2001.

[13] S. D. Crawford, M. P. Couper, and M. J. Lamias, "Web surveys: Perceptions of burden," *Social Sci. Comput. Rev.*, vol. 19, no. 2, pp. 146–162, 2001.

[14] S. J. Sills and C. Song, "Innovations in survey research: An application of web-based surveys," *Social Sci. Comput. Rev.*, vol. 20, no. 1, pp. 22–30, 2002.

[15] G. W. Yun and C. W. Trumbo, "Comparative response to a survey executed by post, email, and web form," *J. Computer-Mediated Commun.*, vol. 6, no. 1, 2000.

[16] D. Bowker and D. A. Dillman, "An experimental evaluation of left and right oriented screens for web questionnaires," in Annu. Meeting Amer. Assoc. Public Opinion Res., Portland, OR, May 2000.

[17] C. D. Redline and D. A. Dillman, The Enfluence of Auxiliary, Symbolic, Numeric, and Verbal Languages on Navigational Compliance in Self-Administered Questionnaires. Pullman, WA: Washington State Univ., 1999. Unpublished manuscript.

[18] M. P. Couper, R. Tourangeau, and D. M. Steiger, "Social presence in web surveys," in Assoc. Comput. Mach. CHI 2001 Conf., Seattle, WA, Mar./Apr. 2001.

[19] D. A. Dillman, R. D. Tortora, J. Conradt, and D. Bowker, "Influence of plain vs. fancy design on response rates for web surveys," in *Annu. Meeting. Amer. Statist. Assoc.*, 1998.

[20] B. K. Kaye and T. J. Johnson, "Research methodology: Taming the cyber frontier," *Social Sci. Comput. Rev.*, vol. 17, no. 3, pp. 323–337, 1999.

[21] J. Baron and M. Siepmann, "Techniques for creating and using web questionnaires in research and teaching," in *Psychological Experiments on the Internet*, M. H. Birnbaum, Ed. San Diego, CA: Academic, 2000, pp. 235–265.

[22] D. Bowker, "Constructing the client-computer interface: Guidelines for design and implementation of web-based surveys," Washington State Univ., Pullman, WA, Summary Report no. 95-59, 1999.

[23] A. Frick, M. T. Bachtinger, and U.-D. Reips, "Financial incentives, personal information and drop-out rate in online studies," in *Current Internet Science—Trends, Techniques, Results*, U.-D. Reips, B. Batinic, W. Bandilla, M. Bosnjak, L. Graf, K. Moser, and A. Werner, Eds. Zurich, Switzerland: Deutsche Gesellschaft für Online-Forschung, 1999. [Online]. Available: http://www.dgof.de/tband99/pdfs/a_h/frick.pdf.

[24] M. Fuchs, M. P. Couper, and S. E. Hansen, "Technology effects: Do CAPI or PAPI interviews take longer?," *J. Official Statist.*, vol. 16, no. 3, pp. 273–286, 2000.

[25] A. Parush, R. Nadir, and A. Shtub, "Evaluating the layout of graphical user interface screens: Validation of a numerical computerized model," *Int. J. Human-Computer Interaction*, vol. 10, no. 4, pp. 343–360, 1998.

[26] D. A. Dillman, C. D. Redline, and L. Carley-Baxter, "Influence of type of question on skip pattern compliance in self-administered questionnaires," in Amer. Statist. Assoc. Annu. Meeting, Baltimore, MD, Aug. 1999.

[27] D. A. Dillman, L. Carley-Baxter, and A. Jackson, "Skip-pattern compliance in three test forms: A theoretical and empirical evaluation," Washington State Univ., Pullman, Technical Report 99-01, 2001.

[28] N. Schwarz, "What respondents learn from questionnaires: The survey interview and the logic of conversation," *Int. Statist. Rev.*, vol. 63, no. 2, pp. 157–177, 1995.

[29] G. Francis, I. Neath, and A. Surprenant, "The cognitive psychology online laboratory," in *Psychological Experiments on the Internet*, M. H. Birnbaum, Ed. San Diego, CA: Academic, 2000, pp. 267–284.

[30] M. Bosnjak and T. L. Tuten, "Classifying response behaviors in web-based surveys," *J. Computer-Mediated Commun.*, vol. 6, no. 3, 2001.

[31] D. A. Dillman, S. Caldwell, and M. Gansemer, Visual Design Effects on Item Nonresponse to a Question About Work Satisfaction That Precedes the Q-12 Agree-Disagree Items. Pullman, WA: Washington State Univ., 2000. Unpublished manuscript.

[32] C. DeRouvray and M. P. Couper, "Designing a strategy for reducing 'No Opinion' responses in web-based surveys," *Social Sci. Comput. Rev.*, vol. 20, no. 1, pp. 3–9, 2002.

[33] D. Heerwegh and G. Loosveldt, "Web surveys: The effect of controlling survey access using PIN numbers," *Social Sci. Comput. Rev.*, vol. 20, no. 1, pp. 10–21, 2002.

[34] R. A. Dennis and S. S. Gambhir, "Internet Question and Answer (iQ&A): A web-based survey technology," *IEEE Trans. Inform. Technol. Biomed.*, vol. 4, no. 2, pp. 116–125, 2000.

**Jake Burkey** is a research analyst at the Center for Economic Development, Department of Applied Economics and Statistics, University of Nevada, Reno. He holds a B.S. in Biochemistry from the University of California, Davis, and a M.S. in Resource and Applied Economics from the University of Nevada, Reno. His research interests include the development and administration of web surveys and the design and application of econometric models.

**William L. Kuechler** is an associate professor of Computer Information Systems, University of Nevada, Reno. He holds a B.S. in Electrical Engineering from Drexel University, Philadelphia, PA, and a Ph.D. in Computer Information Systems from Georgia State University, Atlanta. His research interests include interorganizational workflow and coordination and the organizational effects of interorganizational systems. He has published in IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, *Decision Support Systems, Information Systems Management, Annals of Information Technology Cases*, the proceedings of WITS, HICSS, and other international conferences and journals. Dr. Kuechler is a member of the ACM.