

OnQ: An Authoring Tool for Dynamic Online Surveys

Roy P. Pargas¹, James C. Witte², Laura Brand¹, Catherine Hochrine¹, Margaret Staton¹

Department of {¹Computer Science, ²Sociology}, Clemson University

Clemson, SC 29631

{pargas, jwitte, aklaura, cashiel, mestato}@clemson.edu

Abstract

Internet growth and activity has been increasing almost exponentially over the last decade. Today, millions of people all over the world use the Internet for a variety of activities, such as email and instant messaging, shopping, trading stocks, research, and many others. Of these, one of the fastest growing is online surveys. Because of its global reach, the Internet is the perfect medium for conducting surveys. In recent decades surveys have become a widely used tool to measure individuals' opinions, preferences and behavior. A web-based survey is a very effective way to reach large numbers of people from all over the world compared to paper, in-person, or telephone surveys. A survey authoring tool called *OnQ* makes the creation and deployment of web surveys intuitive and efficient, providing tools and features that commercially available survey authoring tools do not. This paper gives an overview of *OnQ*, provides details about each of its four main components, and describes currently active surveys produced by *OnQ*. The paper concludes with a summary and a view of plans for the future.

1 Introduction

To many Americans the use of the Internet has become so much a part of their lives that they hardly recognize the remarkable concept behind it. However, ten years ago the concept of using a computer to connect to a network of other computers from which one could gather information and perform every day tasks would have been unfamiliar and even bizarre to almost everyone. Before 1990 the only manifestations of Internet activity existed in academic and research fields and consisted primarily of email and file transfers, but by the mid nineties, the phenomenon known as the World Wide Web had exploded with commercial activity which surpassed that of the research world. The web became "point-and-click" and "user-friendly" with graphical web sites and easy to use web search engines. The number of accessible web sites also increased rapidly. In the early nineties only about 300,000 host sites existed, but less than ten years later, in January of 2000, over 72 million sites could be found [1]. The Internet has proven itself to be "one of the fastest developing technologies the world has ever known" [4, p. 146].

Currently, the percentage of Americans (U.S.) who use the Internet is greater than 54% (estimated for September 2001) [2]. This is about 143 million people, an increase

of twenty-six million over the preceding thirteen months. Today computers are located in our offices, schools, libraries, and homes. Other countries have been experiencing the same increase in use. An estimate for February 2001 estimates Sweden's population usage to be over 58%. Denmark (53.6%), the Netherlands (53.3%), and Finland (51.1%) have the next highest percentages. The United Kingdom has close to forty percent (39.8%), and France and Germany have 24.5% and 29.2% respectively. The average for the EU is just over thirty percent (31.1%).

The Internet has changed the way all of us interact with the world around us by altering the way information is gathered and exchanged. Unlike the non-interactive technologies of years past (newspaper, radio, television) that helped to define nations by informing citizens of national policy and culture and inspiring a sense of national pride, the Internet has had a primarily *globalizing* effect on the people of the world. The web provides an open forum for expression, and by eliminating the time (and money) involved with cross-global communication, it connects and unites people the world over [3]. This changes the way we think about the world around us. Graham writes on the subject, "The *worldwide* Web is properly so-called. It has brought into existence a degree of internationalism which is without precedent. By subverting national boundaries it calls into question the power of the state as the dominant force in social life and thus permits the reconfiguration of human communities in line with chosen grounds" [4, p. 38].

The rise of the Internet has also benefited various existing commercial and academic fields. One such field is survey research. Surveying is regularly used by businesses to obtain feedback about products or services from customers. Academic researchers use surveys to track a wide range of attitudes and behavior. Surveying on the Internet has several advantages over the traditional paper or in-person forms. Web surveying offers lower costs in terms of both time and money [5]. The traditional paper survey requires the costs of survey publishing, survey distribution, respondent reminding, survey collection, data entry, feedback publication, and feedback distribution. In-person and telephone surveys require an active interviewer for each respondent.

Online surveying means less data entry time with fewer mistakes for both the respondent and the surveyor (data entry person). It means that participants can receive

individualized feedback during the survey (which has been shown to motivate a participant to complete a survey) and can be honest without the self-consciousness that comes from answering in front of an interviewer [6].

Online surveying also allows the use of various question formats and the use of images, sounds, or other media to prompt responses. By dynamically evaluating a respondent's answers an online tool can avoid asking irrelevant questions and can perform checks to make sure questions are not skipped and the answers provided by a respondent are appropriate (e.g., not using letters where only numbers are meaningful), and administrators have immediate access to results of the survey enabling them to track the data collection process [5].

With so many diverse groups of people on the Internet today, online surveying already has an important role in business and will continue to have such a role in the future. A tool to create online, interactive surveys is the focus of this paper.

2 Background

In 1998, researchers from Northwestern University led scholars around the world to investigate the question *How does your environment affect who you are?* Funded by the National Geographic Society (NGS), their efforts resulted in *Survey2000*. The survey was hosted on the National Geographic website from October through December of 1999. Some of the questions presented to the over 80,000 respondents dealt with how often they moved, how strongly they felt about their communities, how much they used the Internet, which social groups they had ties with, and what they preferred in terms of food, music, and authors. The survey was developed on a Unix work station using CGI scripts. Questions, answers, and the ability to proceed to one question versus another, as determined by a given question's answer, were all hard coded in the CGI scripts ([7,8]).

Because of the positive response to *Survey2000*, NSF agreed to fund *Survey2001*¹. Led by researchers from Clemson University, sociologists around the world focused on Internet activity and how use of this and other information technology affects society. Specifically, the researchers wanted to know how the Internet might *redefine* one's sense of geography. To assess this proposition, questions pertaining to community, culture, and conservation were incorporated into the survey. Launched in October, 2001, the survey was available in four languages (English, Spanish, German, and Italian) [7,8] and collected responses through September, 2002.

Implementing *Survey2001*, however, was difficult. The survey questions and possible answers were entered into the database, to a large extent, by hand, and the effort required to accurately insert over 2000 questions and 1500 answers (for each language), was high. Some errors

took hours to locate. The reliance on the data entry person, and the need for meticulous precision, was extremely high. Furthermore, different sections of the questions of the survey were written by different sociologists (in part because the survey was to be duplicated in several different languages), but the questions and answer options could ultimately only be entered by computer scientists familiar with database SQL statements. During the development of the German version of the survey, for example, German translators received copies of English questions, translated them into German, and returned the translated questions in an Excel spreadsheet to the computer scientists who then wrote programs that read from the spreadsheets and inserted the questions into the database. The German translators then checked the accuracy of the survey content (on the Web), and recommended changes and corrections, which had to be made manually (again by a computer scientist). Entering, checking, and testing of the entire survey took hundreds of hours. In total, it took eight sociologists, three computer scientists, and six Clemson University and National Geographic personnel over six months to enter the data and develop the software to create *Survey2001*.

3 Problem Statement

The need for a survey authoring tool was clear. The problem at the time was that none of the tools available when *Survey2001* was being developed satisfied all of the requirements of *Survey2001*. Some of the authoring tools considered are described here.

Web Surveyor [9] is available for PCs running Win95 and later operating systems, or it can be used from their server. The system can handle surveys, versatile branching, different background colors, and different text options.

WWW Survey Assistant is Web-accessible, and works with Win95 or later. Because it is Java-based, it can be run on any machine, including Solaris, OS/2, and Macs. All project files are saved on their server and provides multiple text fonts. The software claims to randomize questions [10] and to handle complex skip patterns but details are sketchy.

Hosted Survey [11] is entirely web-based. It handles conditional branching where the answer to a single question can branch to a series of questions, and it can randomize questions. The tool allows questions to be grouped into sections.

Survey Said for the Web [12] is Web-accessible and works with Win95 or later. This tool allows for single, linear skips only. One feature of this authoring tool is the use of keyword association for easily identifying questions, and also a search feature for the questions where it can search by word or phrase.

Finally, *Sawtooth WebSensus* [13] is Web-accessible and works with Win95 or later. This tool handles branches of any complexity, and can randomize the answers,

¹ <http://johnryan.clemson.edu/ngm/servlet/Page1>

questions, or blocks of questions. A table summarizing these authoring tools is displayed in Figure 1.

The *Survey2001* developers decided that the ideal tool would have a graphical user interface allowing a developer to enter questions and answer options which would be automatically be written by the program into a database. It would also provide a way for the user to create blocks or groups of questions and add transitions among these blocks (*transitions* instruct the survey presentation tool as to which group of questions to present *next* after the current group is answered, see Section 4). Question blocks and transitions would be stored in the database. The developer would also be able to present individual questions in several different layouts and would be able to group questions with identical radio button answers options a compact radio button table. The tool would also allow questions to be randomly presented with a choice of several randomization functions. The program would allow access to external databases (such as a zip code table) during the execution of a survey. Finally, such a tool could be run standalone as well as on the web where it could be accessed by clients to develop the survey database.

This wish list became the set of goals for *OnQ*.

4 OnQ

OnQ is written in Java (version 1.4) because of Java's cross-platform portability as well as Java's suitability to web applications [14]. *OnQ* accesses a MySQL database (ver. 3.23.47) [15] using a MySQL Connector/J Driver (formally called the MM.MySQL driver) [16]. Pages are served using Apache as the primary web server and the Tomcat (version 3.2.3) from the Apache Group as the servlet engine ([17]). Figure 2 shows an overview of the *OnQ* architecture.

4.1 Architecture

The design of *OnQ* is purposely organized around the steps we anticipate a typical user will take to develop a survey. When *OnQ* starts, the survey author enters a user name and password, and chooses which module and which database to use. If desired, a new database may be created by entering a name for the new database.

With a new database, the first module the survey author will want to work with is the *Question Editor* which enables the author to enter and modify questions and answers. The author selects answer option formats (radio buttons, drop-down menus, checkboxes, textboxes, or text areas) and media components (image, audio, video, or animation). The *Question Editor* allows the association of a key with each question to facilitate searching for a question at a later time and provides a mechanism for the author to group questions into categories, enabling him or her to work on one related set of questions at a time. [18]

The next step is to form logical subgroups (within categories) called *question blocks*. This function is provided by the *Sequence Developer* which enables the

author to specify the order in which question blocks are presented (i.e., by creating Boolean expression-controlled transitions between pairs of blocks). This is explained in greater detail in Section 4.2 below. The author may also specify that questions within a question block are to be presented in some random fashion. Error checks within this module eliminates the possibility that the *Presentation Manager* will have to deal with errors such as incorrect transitions or question blocks, or questions not included in the survey.

After the creation of question blocks and transition sequencing rules, the author can use the *Formatter* module to specify the layout of a question or of a question block. By default all questions within a question block are presented on a single web page when the survey is deployed. The *Formatter* module, however, allows an author to specify that questions in a given question block be displayed on separate pages. Another option available in this module is to create tables of radio button questions that are within one question block. [21]

The *Question and Block Formatter* provides the developer with a number of layout options. For each question, media components may be placed above, beside, or below the question text, question prefixes and suffixes may be joined to or separated from the question root.

Using the *Block Formatter*, the developer may choose to present a large question block to the client in a series of pages with a small number of questions on each page. This module also provides an option to display a sequence of radio button questions as a single radio button table.

The interaction among the modules (*Question Editor*, *Sequence Developer*, and *Formatter*) is purposely kept simple and limited. Each module writes data directly to the database as changes are made and other modules update their data from the database each time they are accessed. A user can access any of the modules in any order through tabs at the top of the program frame. However, the *Question Editor* must be used to enter at least one question and one set of answers before the other modules can have some data with which to operate.

After questions and answers are entered, images are assigned, question blocks and block sequencing rules are defined (transitions), and the questions are formatted, the developer can use the fourth component of *OnQ*, the *Presentation Manager*, to view the completed survey. The *Presentation Manager* is a Java servlet that extracts the survey information directly from the database and uses this information to serve survey web pages. All of the information it needs is stored in the database. The *Presentation Manager* presents questions, answers, and accompanying media components entered with the *Question Editor*, in the order specified by the author in the *Sequence Developer*, and according to the question and block formats specified in the *Formatter* module. During development of the survey, the author's *local* browser can present the pages exactly as they would be

presented on the Web. Additions and corrections can be made directly to the database using *OnQ* and the author can view the changes immediately.

The author can deploy the survey on the Web when he or she is satisfied that the survey is complete and correct.

4.2 Question Blocks

The concept of question blocks is central to the correct presentation of a survey and requires more explanation. Certain policies were set for questions within a block. First, it was decided that (a) if a question block is not randomized, then every question within a block is to be presented in the user-specified order, before the survey moves to another block; and (b) if the questions within a block *are* randomized in some way then the questions selected by the randomization function are to be presented in the order specified by the function, before the survey moves to another block. Second, adding a question to a question block involves associating one instance of a question with the order within that one question block. And third, question block sequencing (as defined by question block transitions) determines the order in which question blocks are presented in the overall survey.

To implement this, the *Sequence Developer* initially creates a two distinguished blocks called START and END. Both blocks are empty (i.e., contain no questions) and START, by default, points to END. The author must change the default transition from START to a non-empty question block (otherwise the survey will begin and end without asking any questions). This non-empty block contains the first questions to be presented in the survey.

From the first non-empty question block, the author can specify the *conditions* under which control passes to other question blocks. This transfer of control is called *transitioning*, which is implemented with the use of user-defined 4-tuples:

(source, expression, destination, seqnum)

where *source* is the source block, *expression* is a Boolean expression to be evaluated for each client at survey deployment time, *destination* is the destination block, and *seqnum* is a sequence number imposing a linear order on the expressions associated with block *source*.

Consider the following set of 4-tuples prepared by the *Sequence Developer*. Assume that question 12 of block 1 is the question: "Are you 18 or older?" Assume further that answer A2 is "No" and answer A3 is "Yes".

- (1, $Q12=A2$, 2, 1)
- (1, $Q12=A3$, 3, 2)
- (1, *default*, 4, 3)

Interpreting these tuples, the *Presentation Manager* will steer minors (clients under 18) to block 2 and adults to block 3. Clients who do not answer will be presented questions from block 4. Expressions are evaluated in *seqnum* order. Block 4 may contain age-neutral questions

or an error statement requesting that the client go back and answer questions on the previous page.

Note that an expression is an arbitrary Boolean expression and may be as complex as the author chooses. It is assumed that all question variables used in the expression represent questions already presented to the client. Otherwise, the expression evaluator of the *Presentation Manager* will report that the expression cannot be evaluated and the survey cannot proceed.

To model transitioning among blocks, it is natural to use a deterministic finite state automaton or acceptor (DFSA) described as follows:

$$M = (Q, \Sigma, \delta, q_0, F)$$

where

M = the automaton and

Q = set of internal states,

Σ = set of symbols (input alphabet),

q_0 = initial state,

F = set of final states, and

$$\delta = Q \times \Sigma \rightarrow Q$$

(δ is a total function called the transition function which gives the next state in terms of current state or configuration and input symbol; a move is a transition by the automaton from one configuration to another).

Transition graphs are traditionally used to visualize and represent finite state automata. In a transition graph, vertices represent states, edges represent transitions, labels on vertices represent names of states, labels on edges represent current values of input symbol. For example an edge labeled with (q_0, a) represents the transition function $\delta(q_0, a) = q_1$ (usually read as, "from state 0 with the input of an 'a,' move to state 1") [19].

The *Presentation Manager* interprets the information recorded in the database in such a way that it essentially moves through the state diagram by evaluating Boolean expressions and transitioning to target question blocks accordingly.

One difference, though, is that in a traditional DFSA only one transition (one edge) can be taken based on a given input. In the *OnQ* model there may be multiple transitions that evaluate to TRUE when the survey is taken. Each transition from a question block is numbered from 1 upward. Transitions from one question block are evaluated in order until the first expression evaluating to TRUE is found, and the corresponding transition to a next block is taken. Also, the last transition must be a default transition that *always* evaluates TRUE, guaranteeing that a transition will be taken.

4.3 Graphical Representation of Question Blocks

A useful feature of *OnQ* is that it can present to the author a graphical view either of a single question block (Figure 3) with all transitions to and from the block, or of the

entire survey (Figure 4). To implement this, a software tool called *dot* contained in a package of graph drawing tools called *Graphviz* from AT&T Research [20] is used. *Dot* is a preprocessor which reads a script file description of a graph and draws the graph in one of several formats, including GIF, JPG, and PS. The script and graph are developed by *OnQ* and by *dot*, respectively, in real-time, reflecting all changes made by the author to question blocks and transitions up to that time. To our knowledge, no other online survey authoring software offers anything analogous to this.

4.4 Error Checking

Using the question block model of function also allowed us to develop some algorithms to check user input for possible errors. One function the *Sequence Developer* provides is alerting the user if a question is not a member of any question block or a question block is not part of the survey's network of blocks and transitions.

5 Summary and Results

OnQ makes it easy for a survey author to create, edit, and format questions and answer options for a survey in a user-friendly Graphical User Interface (GUI). It allows the author to define question blocks which will implement skip patterns based on the respondent's answers to some of the questions. For the transitions between question blocks, the skip patterns are represented by arbitrarily complex Boolean expressions.

It is written in Java and can run on any platform with the Java Virtual Machine. It has been implemented on a Windows PC, on an Apple Macintosh, and on a Sun Solaris system.

OnQ allows for the insertion of audio, image, and video components for any given question. *OnQ* also has the ability to be created in any language and/or multiple languages; it can categorize questions into logical groups; and it has keyword association for easily identifying the questions.

OnQ checks to see if there are any questions not included in any question blocks; and if there are any dangling question blocks, i.e. if there are any question blocks to which there are no transitions.

OnQ makes it easy for the survey author to toggle between the four modules while developing a survey. It gives the survey author the ability to see a graphical view of the survey including the question blocks and transitions. It also allows the survey author to view the presentation of the survey exactly as it will appear to the respondents taking the survey, allowing the author to fine-tune the survey's format along the way.

The separation of the tool into four main modules helps the user to differentiate between the tasks of creating questions and answers, forming question blocks and transitions among them, and formatting questions and blocks. *OnQ* also provides easy ways to switch between

modules. This also enables the user to choose either a linear method of entering questions or a more random method, as the author prefers.

Overall, *OnQ* fulfills nearly all of our original goals for a survey authoring tool. First, *OnQ* creates and populates all the necessary database tables with enough data to present a large and dynamic survey. Secondly, we feel that *OnQ* is easy to use. *OnQ* offers non-technical users the simplicity of a point-and-click interface.

Third, the *OnQ* authoring tool has already been used to develop three currently active online surveys. *Diversity2002*² is the second in a series of four annual surveys. Students in the Clemson University Class of 2005 are being surveyed every year and asked questions related to race, diversity, and race relations. This four year study hopes to come up with recommendations on how to improve campus life by reducing or eliminating situations that may create racial tensions.

*UnitedWay2002*³ is collecting opinions and ideas from members of the Greenville, SC community regarding local community strengths and issues. The information is to be used to develop Community Impact Agenda, which will be a road map for the community on how to meet its most urgent needs. This effort is led by the United Way of Greenville County, SC with partners representing various sectors of the community. A report is expected in January 2003, with the Community Impact Agenda complete by the end of May 2003.

*IE2002*⁴ is being conducted by the Clemson University Industrial Engineering Department and attempts to survey and compare Industrial Engineering curricula across the country. The primary objective is "the development of a new scaleable and deployable industrial engineering baccalaureate-degree renewal model. This model will be designed to permit scaling up from an information technology kernel to a fully integrated industrial engineering undergraduate curriculum."⁵

Moreover, the Center for Advancement of Marketing and Social Sciences at Clemson University has decided to work with *OnQ* to develop online surveys for a 6-month trial period.

6 Future Work

Several improvements are planned for *OnQ*. First, the code will undergo continued review and testing. Errors will be corrected and the modules will be streamlined. The goal is to develop a CD with a complete and installable standalone system.

² <http://johnryan.clemson.edu/Diversity2002/servlet/Page1>

³ <http://johnryan.clemson.edu/UnitedWay/servlet/Page1>

⁴ <http://johnryan.clemson.edu/IE2002/servlet/Page1>

⁵ <http://www.ces.clemson.edu/~agramop/currirennewal.htm>

Second, a web-version of the authoring tool is planned. The developer will be able to click on a URL and use the authoring tool remotely. The database will be stored, by default, on the server on which *OnQ* resides. The developer decides when to release the survey for public use. Until that time, only the author can modify and view the survey being developed.

Third, a DVD training program is planned for development. The DVD will accompany the software and will help a novice user work through the survey authoring process including instructions in the proper construction of survey questions, pros and cons of the different answer options, and providing a large number of examples on how to use the various features of *OnQ*.

Finally, there are plans to develop *OnX*, short for *Online eXercises/eXamination*. *OnX* is the next step beyond *OnQ*, and uses *OnQ*'s architecture and database design. *OnX* will be an authoring tool that allows a teacher to develop web-based learning and assessment instruments. The overall goal of *OnX* is to help pre-K through 12th grade teachers develop software tools and a strategy for coordinating and delivering a standards-based curriculum. For example, *OnX* may be used to create *Math8*, a learning and assessment instrument for 8th grade math. *OnX* will have to extend *OnQ* in many ways, such as: assign to each question a point value, as well specify the correct answer(s); randomize answer choices; include the ability to enter math symbols; provide a grade book and grade reporting functions; and many more.

7 Acknowledgements

Support for this work has been provided by the National Science Foundation (NSF-ITR/Soc (Award # 0082750), the Clemson University Office of Access and Equity, the United Way of Greenville County, and the Clemson University Department of Industrial Engineering.

References

1. Nunes, M. 2001. Ephemeral cities: postmodern urbanism and the production of online space. In *Virtual Globalization: Virtual Spaces/Tourist Spaces*, D. Holmes, Ed. Routledge, New York.
2. U.S. Department of Commerce, Economics and Statistics Administration, National Telecommunications and Information Administration. Feb. 2002. A Nation Online: How Americans Are Expanding Their Use of the Internet.
3. Stratton, J. 1997. Cyberspace and the Globalization of Culture. In *Internet Culture*, D. Porter, Ed. Routledge, New York.

4. Graham, S. 2001. Telecommunications and the future of cities: debunking the myths. In *Virtual Globalization: Virtual Spaces/Tourist Spaces*, D. Holmes, Ed. Routledge, New York.
5. Schmidt, W. C. 1997. WorldWide Web Survey Research: Benefits, Potential Problems, and Solutions. *Behavior Research Methods, Instruments and Computers* 29, 2, 274-279.
6. Jenkins. C.R. and Dillman, D.A. 1997. Towards a Theory of Self-administered Questionnaire Design. In *Survey Measurement and Process Quality*, L. Lyberg, P. Biemer, M. Collins, E. DeLeeuw, C. Dippo, N. Schwarz, and D. Trewin, Eds. John Wiley, New York.
7. Jaganathan, K. 2002. *Dynamic Web based Survey: Architecture and Implementation*. M.S. research paper. Dept. Computer Science, Clemson University, Clemson, SC.
8. Pargas, R.P., Witte. J.C., Jaganathan, K., Davis, J.S. *Database Design for Online Dynamic Surveys: A Case Study*. Submitted to ITCC2003, Las Vegas, NV.
9. WebSurveyor Corporation. 2002. *Web Surveyor*. http://www.websurveyor.com/prof_intro.sap
10. S-Ware. 1996-1998. *WWW Survey Assistant*. <http://or.psychology.dal.ca/~wcs/hidden/home.html>
11. Hostedware Corporation. 2002. *Hosted Survey*. <http://www.hostedsurvey.com/Spotlight.htm>
12. Marketing Masters. 2002. *Survey Said For the Web*. <http://www.surveysaidfortheweb.com>
13. Sawtooth Technologies. 2002. *Sawtooth*. <http://www.sawtooth.com>
14. Sun's Java, <http://java.sun.com/>
15. MySQL, <http://www.mysql.com>
16. Connector/J-Driver, <http://www.mysql.com/downloads/api-jdbc.html>
17. Apache Http Server, <http://httpd.apache.org/>
18. Hochrine, C. *The Development of the Question Editor for OnQ, a Survey Authoring Tool*. M.S. research paper. Dept. of Computer Science, Clemson University, Clemson, SC.
19. Linz, P. 1996. *An Introduction to Formal Languages and Automata*, 2nd edition. D.C. Heath and Company, Lexington, Massachusetts.
20. AT&T *Graphviz* package, <http://www.graphviz.org>
21. Staton, M. 2002. *The Development of the Formatting Module for OnQ, a Survey Authoring Tool*. Undergraduate honors thesis in preparation. Clemson University, Clemson, SC.

Figure 1. Table summarizing web survey authoring tools.

	WS	SA	HS	SSW	SWT	OnQ
Win95+	*	*		*	*	*
Other machines		*				*
Images & audio	*	*		*	*	*
Question categories			*			*
Branching	*	*	*	*	*	*
Randomization		*			*	*
Multi-language	*					*
Keyword association				*		*
Different backgrounds	*					*
Different text	*	*				*

Web Survey (WS), Survey Assistant (SA), Hosted Survey (HS), Survey Said on the Web (SSW), Sawtooth (SWT) and OnQ

Figure 2. Architecture of OnQ

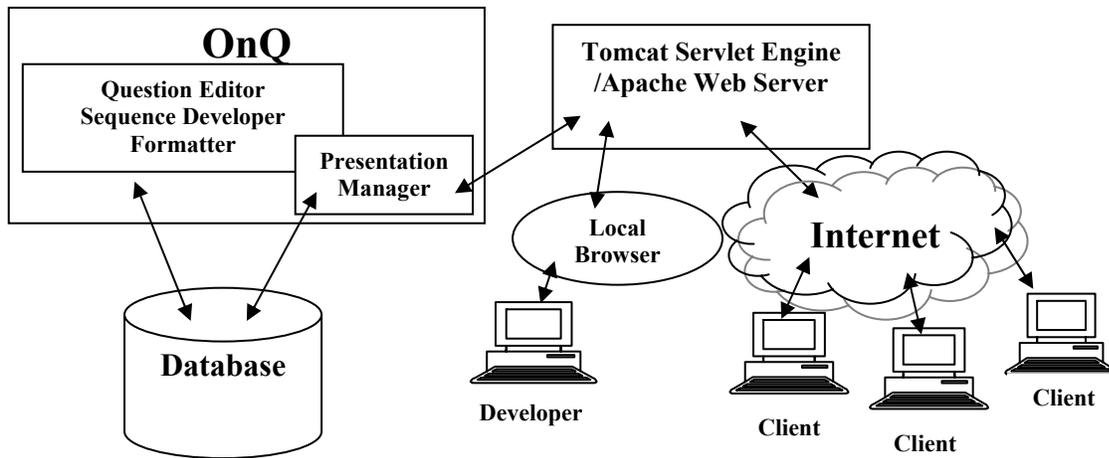


Figure 3. Transitions to and from a single question block.

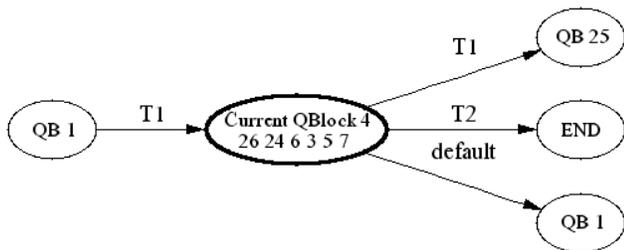


Figure 4. View of an entire survey.

