Technical Report

Remote Usability Testing Methods a la Carte

José C. Castillo
eonBusiness Corporation
7430 East Caley Ave. Suite 200
Centennial, Colorado 80111 USA
jose.castillo@eonbusiness.com

H. Rex Hartson
Department of Computer Science
Virginia Tech
Blacksburg, VA 24061

Technical Report TR-07-05
Department of Computer Science
Virginia Tech
Blacksburg, VA 24061

Remote Usability Testing Methods a la Carte

José C. Castillo
eonBusiness Corporation
7430 East Caley Ave. Suite 200
Centennial, Colorado 80111 USA
jose.castillo@eonbusiness.com

H. Rex Hartson
Department of Computer Science
Virginia Tech
Blacksburg, VA 24061

Abstract

Although existing lab-based formative usability testing is frequently and effectively applied to improving usability of software user interfaces, it has limitations that have led developers to turn to remote usability evaluation methods (RUEMs) to collect formative usability data from daily usage by real users in their own real-world task environments.

The enormous increase in Web usage, where users can be isolated and the network and remote work setting become intrinsic parts of usage patterns, is strong motivation for supplementing lab-based testing with remote usability evaluation methods. Another significant impetus for remote evaluation is the fact that the iterative development cycle for any software, Web application or not, does not end with initial deployment. We review and informally compare several approaches to remote usability evaluation with respect to quantity and quality of data collected and the effort to collect the data.

Keywords

Remote usability evaluation, evaluation method, remote usability method, user-reported critical incident method, critical incident, usability testing.

Background

The remote and distributed location of users precludes direct observation of their usage, and transporting users to developer locations (or vice versa) for usability testing can be very costly. Also, as the remote work setting has become an intrinsic part of usage patterns, the users' work context can be difficult or impossible to reproduce in a laboratory setting. But real usage can be essential to ecological validity that is sometimes missing within lab-based usability testing [Thomas & Kellogg, 1989].

These barriers to effective lab-based testing have led researchers and practitioners to seek ways to extend usability evaluation beyond the laboratory by developing the concept of remote usability evaluation, typically using the network itself as a bridge to make an interface evaluation connection to a broad range of users in their natural work settings. Fortunately, deployment of a Web site or other application creates an additional source of usability data associated closely with real task performance, not available to be captured locally in a lab. Remote usability evaluation methods (RUEMs) can offer solutions by providing low cost data gathering, cost reduction in data analysis, and usability data from real-world post-deployment usage.

Defining Users and Evaluators

We distinguish the two important roles in remote usability evaluation by using the term "user" to refer to the remote end-user and the term "evaluator" to refer to the developer role responsible for usability, someone trained in usability methods. A RUEM is defined as a technique for formative usability evaluation where evaluators are separated in space and/or time from users [Hartson & Castillo, 1998], as shown in Figure 1. The term remote is used relative to the evaluators and refers to users who are not at the location of evaluators. Similarly, the term local refers to the location of the evaluators.

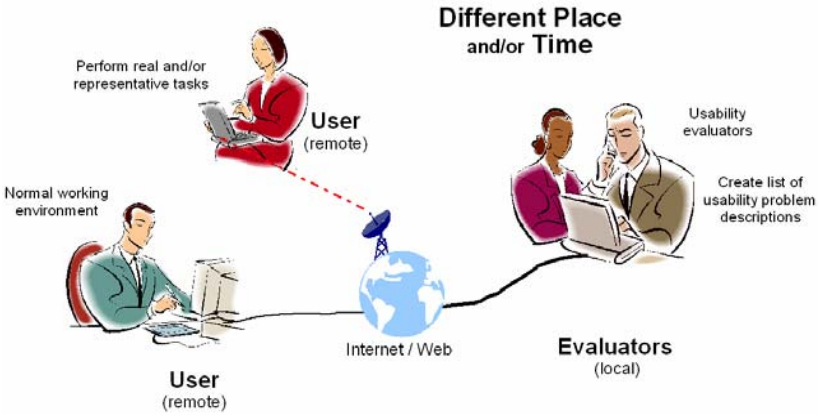


Figure 1. Scenario of a remote usability evaluation session

Traditional Lab-Based Usability Evaluation

Traditional laboratory-based usability evaluation is often used as a practical yardstick for comparison with most new methods. Lab-based evaluation is usually considered "local evaluation" in the sense that the user and evaluator are in the same location at the same time. Data collected can be both quantitative (e.g. task performance time and error rates) and qualitative (e.g., critical incident descriptions and verbal protocol), the former to assess the level of usability achievement and the latter to identify usability problems and their causes within the interface design [Hix & Hartson, 1993].

Remote Usability Evaluation Techniques

To compliment traditional lab-based evaluation, several different kinds of RUEMs have been developed for conducting usability evaluation at a distance. Each method can be applied independently or combined with another remote evaluation method, or even with traditional lab-based evaluation. Space limitations preclude all but the briefest review of some different types of remote evaluation methods including a very informal comparison of characteristics, advantages and estimated costs, in the following sub-sections.

The RUEMs discussed in this report are:

- User-reported critical incident method
- Remote questionnaire or survey
- Instrumented or automated data collection
- Video-conferencing supported evaluation
- Third-party lab-based usability testing
- Third-party usability inspection

User Reported Critical Incident Method

In this method, users are located in their own working environment and acquire modest Web-based training to identify and report critical incidents occurring in the normal course of on-the-job task performance. Whenever users encounter usage difficulty, they take the initiative (e.g., by clicking on a Report Problem button from their Web browser) to:

- create a structured report on the details of the specific critical incident encountered, and
- create a screen sequence video clip with explanatory audio showing screen activity related to the critical incident and the context of events leading up to it.

The resulting package of usability data—the critical incident report and the screen sequence clip taken together—is called a contextualized critical incident report, sent asynchronously via the network to evaluators to be analyzed into usability problem descriptions that designers use to drive redesign solutions to improve the interaction design.
Because of the vital importance of critical incident data and the opportunity for users to capture it, we developed this method [Hartson and Castillo, 1998] for capturing critical incident data and satisfying the following situational criteria:

- data are captured from day-to-day tasks as performed by real users,
- users are located in normal working environments,
- users self-report their own critical incidents,
- reporting is done within a short time after the problem occurs (i.e., contemporaneous to the usage session),
- no direct interaction is needed between user and evaluator during an evaluation session,
- data capture is cost effective, and
- data are high quality (high value for identifying and fixing usability problems) and relatively easy to translate into usability problem descriptions.

Remote Questionnaire or Survey

There are many variations of RUEMs that use remote questionnaires and surveys to obtain user feedback. Evaluators can send questionnaires to users via email or can give users access to an online questionnaire to gather subjective data about a Web site or application and its interface. In an approach more directly associated with usage details, an application can be

augmented to trigger the display of a pop-up questionnaire to gather subjective preference data about the application and its interface. For example, Alertus® FormSurvey™ [Alertus®, 2006] can be used on a Web site to display a popup questionnaire after a visitor abandons a shopping cart or registration. Remote questionnaires are limited to capturing subjective data based on questions pre-written by developers or evaluators. Thus, in general, critical incidents and other detailed data useful in identifying specific usability problems are not usually collected.

Keynote® [2006] offers a third-party remote usability testing service that combines remote questionnaires with automated data collection. Keynote® employs a large sample of participants (typically 200 to 800 people) who attempt to perform a series of pre-defined "real life" tasks on a target site, using the Keynote Connector (see Figure 2), a small downloadable companion to Microsoft Internet Explorer from their own natural setting [Vividence™, 2002]. Data are automatically compiled within an online interface for the client to review the results.

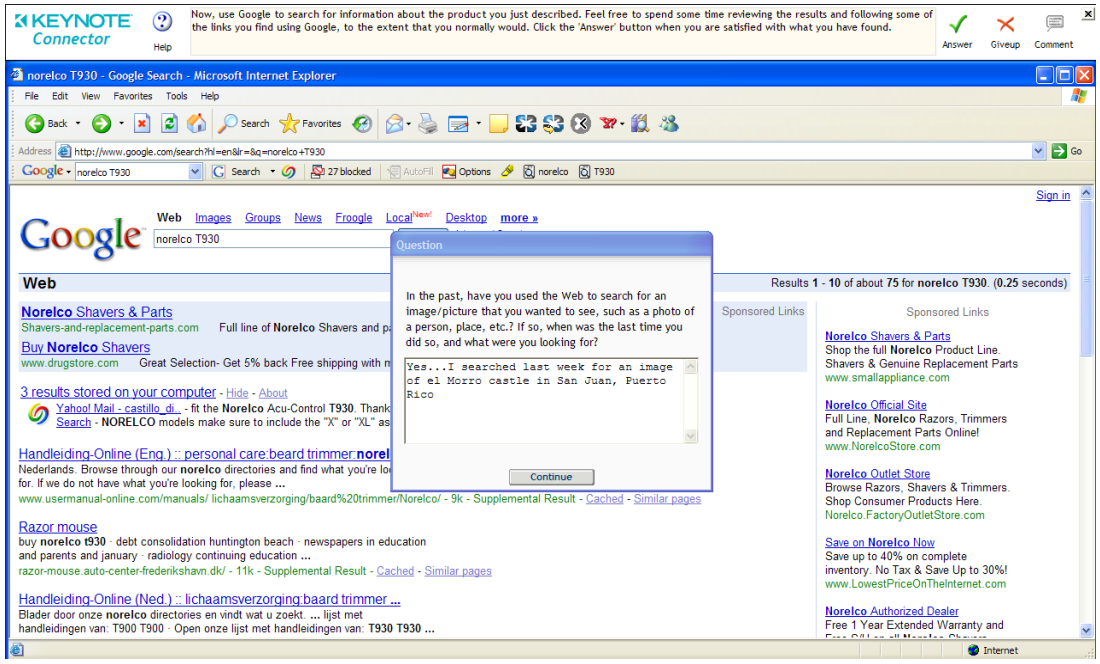


Figure 2. Using the Keynote Connector to evaluate search engines (e.g., Google)

Instrumented or Automated Data Collection for Remote Evaluation

An application and its interface can be instrumented with embedded metering code to collect and return a journal or log of data, such as Internet usage, clickstream (e.g., map of navigation path through site), and mouse movements, occurring as a natural result of usage during on-the-job task performance in the users' normal working environments. Some examples are ErgoLight® Usability Log Analyzer™ [ErgoLight®, 2006]. The logs or journals of data are later analyzed using pattern recognition techniques [Siochi & Ehrich, 1991] to deduce where usability problems have occurred.

This approach has the advantage of not interfering at all with work activities of the user, costs essentially nothing to users or developers to collect data, and can provide automated usability evaluation for certain kinds of usability problems. However, it does not provide (at least not directly) high quality data having the same high value for identifying and correcting usability problems as critical incident data.

Video-Conferencing-Supported Remote Evaluation

Remote evaluation using video-conferencing is an approach that connects evaluators at a usability laboratory to remote users via commercially available collaborative video-conferencing software [Hammontree, Weiler, & Nayak, 1994]. This evaluation method uses telephone or wireless communications and/or the network as a kind of extension to usability lab video and audio facilities [Hartson et al., 1996]. Video data (e.g., user's face, screen-converted image) and audio data (e.g., user comments) are communicated in real time over the network, and perhaps come the closest to the effect of local evaluation. Project budget reductions have reduced travel resources and made this a popular alternative to lab-based usability testing.

Typical tools such as Microsoft® NetMeeting™ [Microsoft®, 2006] or web-based applications such as Raindance® [2006] and Webex® [2006] online meetings can be used, as part of a video-conferencing-supported remote evaluation method, to support real-time application sharing, audio links, and/or file transfer capabilities.

As an alternative, Techsmith's Astoria Project [2006a] (currently in beta version) uses a Web-based service allowing evaluators to remotely connect with remote users, without requiring video conferencing. The screen sequence and audio can be recorded and imported into Morae for analysis and sharing of results into the evaluator's computer (see Figure 3).

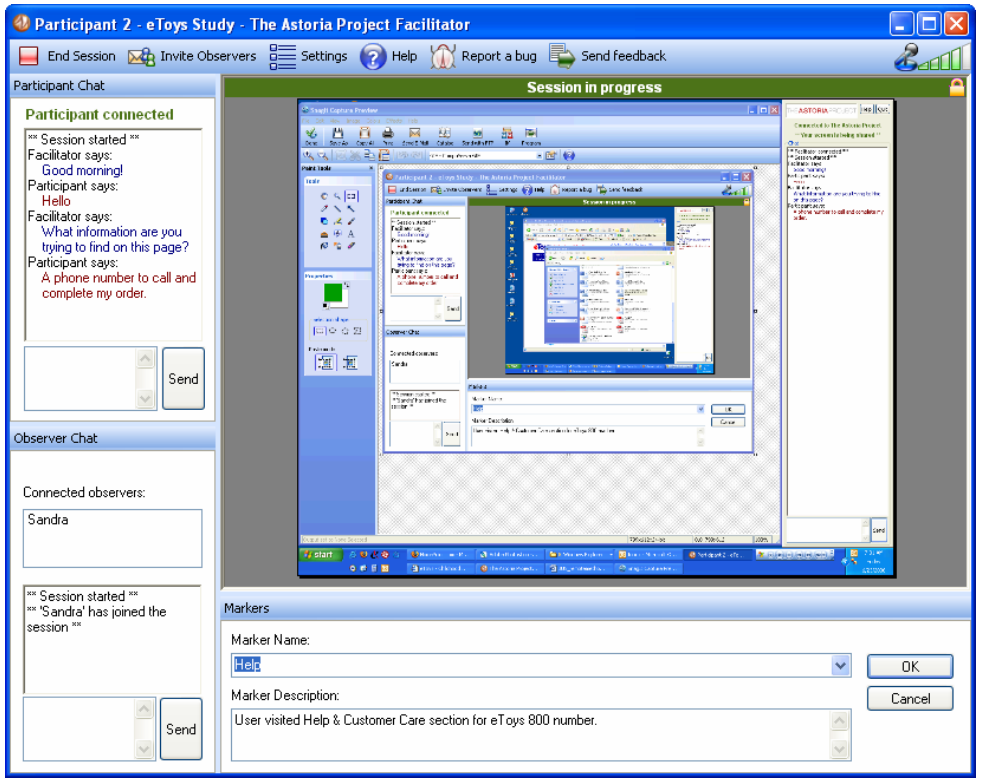


Figure 3. Using the Astoria Project to evaluate an online toy store (e.g., AllAboardToys.com)

Third-Party Usability Inspection

For third-party usability inspection, developers use the network to communicate design documents, software samples, and/or prototypes to remote contractual evaluators, who conduct a usability inspection of the interface without employing users. The evaluation is local to the contractual evaluators but remote from the developers, allowing this method (like third-party lab-based usability testing in the next section) to qualify only technically as a remote method by our definition, but we include these cases in our comparison because of their potential value. The results are returned to the developers via the network.

Third-Party Lab-Based Usability Testing

Developers can do their own usability testing or can hire a third-party service , retaining outside evaluators to conduct evaluation with representative users and

tasks in their own usability lab, remote from the developers, again technically qualifying as a RUEM. Results can include quantitative performance measures, user opinions and satisfaction ratings, usability problem lists, recommendations for application improvement, and even videos of the evaluation sessions for review by the development team (for example, using Morae [2006b]).

Comparing Attributes of Remote Usability Evaluation Techniques

It is useful to distinguish among characteristics of each RUEM to understand how each method works and what can be gained from each method in different situations. The classification made here is not absolute, but relative and representative, and based on insight gained during our work with RUEMs. Although, as with any generalization, there will be variations, exceptions, and special cases that don't fit our descriptions, we wanted to convey the typical characteristics of each method. In that spirit we offer an intuitive comparison of the methods based on our estimates of the following attributes:

- person or role who identifies critical incidents and/or problems during task performance,
- time and place of participation by users and developers,
- type of tasks (the user's own tasks or tasks predefined by the evaluator) and level of user-evaluator interaction,
- type of data gathered,
- quantity of data gathered and type of equipment used for collecting data,
- cost to collect data,
- cost to analyze data and create usability problem descriptions, and
- quality or usefulness of collected data (the value of the data in helping the evaluator and developer identify and fix usability problems).

Traditional laboratory-based usability evaluation (listed as method 7) is included in each discussion as a benchmark method for comparison with the remote evaluation methods.

Person/Role Who Identifies Critical Incidents and Problems

Table 1 indicates the person (or role) who identifies and reports critical incidents or usability problems during the evaluation of the user interface.

Table 1. Characterization of person (role) who identifies and/or reports critical incidents

Remote Usability Evaluation Method	Who identifies and reports critical incidents	
	User	Evaluator
1. User-reported critical incident method	×	
2. Remote questionnaire or survey		
3. Instrumented or automated data collection		×
4. Video-conferencing supported evaluation	×	×
5. Third-party lab-based usability testing*	×	×
6. Third-party usability inspection*		×
7. Traditional lab-based usability testing	×	×

*In third-party cases, the users and evaluators are employed by the third-party service.

RUEM 1 is the only method where users alone identify critical incidents without intervention of an evaluator. At a later time, evaluators receive critical incident reports and analyze them to create a list of usability problem descriptions.

In RUEMs 3 and 6 only the evaluators identify critical incidents, and for RUEMs 4 and 5, both the user and evaluator collaborate to identify critical incidents (as in method 7). By means of RUEM 2, users provide subjective data about their experience with the user interface. Although users *could* use this method to report a critical incident or usability problem and evaluators *could* deduce usability problem information from a user response, this method doesn't usually yield specific critical incident or problem data.

Time and Place of Participation

Table 2 presents relationships of time and place between users and evaluators as they participate (collaborate and communicate) in usability evaluation.

Table 2. Time and place of participation during usability evaluation

Remote Usability Evaluation Method	Time and place of participation		
	Different time, Different place	Same time, Different place	Same time, Same place
1. User-reported critical incident method	×		
2. Remote questionnaire or survey	×		
3. Instrumented or automated data collection	×		
4. Video-conferencing supported evaluation		×	
5. Third-party lab-based usability testing*			×
6. Third-party usability inspection*			
7. Traditional lab-based usability testing			×

*In the third-party case, the users are recruited and evaluators are employed by the third-party service.

As the table shows, usability data collection with RUEMs 1, 2, and 3 occurs remotely within the user's normal work setting (different place from evaluator). User critical incident reports, questionnaires, and log files are analyzed later (different time) by evaluators, who then create a list of usability problem descriptions. RUEM 4 typically involves the user and the evaluator connected in real time (same-time) over the network (different place); 5 and 7 involve local evaluation (same place) with representative users at the evaluators' (third party or developer) usability laboratory (at the same time). Finally, 6 doesn't involve users during an evaluation.

In our current research, the different-time, different-place case was of most interest because the different-time (asynchronous) characteristic means the evaluator does not have to attend the session, making the cost of data gathering low relative to traditional lab-based usability evaluation. Additionally, the different-place (remote) characteristic is less expensive in terms of travel and can mean more realistic qualitative data, since data collection occurs within the user's natural work setting.

Type of Tasks and Level of User-Evaluator Interaction

Table 3 characterizes the kind of tasks for which data can be collected and the level of interaction required between users and evaluators during remote evaluation.

Table 3. Type of user tasks and level of user-evaluation interaction during evaluation

Remote Usability Evaluation Method	Time and place of participation		
	Real tasks, No user-evaluator interaction	Predefined representative tasks, Low-to-significant user-evaluator interaction	Predefined representative tasks, No user-evaluator interaction
1. User-reported critical incident method	×		
2. Remote questionnaire or survey	×		×
3. Instrumented or automated data collection	×		
4. Video-conferencing supported evaluation		×	
5. Third-party lab-based usability testing*		×	
6. Third-party usability inspection*			×
7. Traditional lab-based usability testing		×	

We define "real" tasks as all the usual interactions users perform while conducting their normal, everyday activities (e.g., at work or home), taking in consideration any context that they might have (e.g., interruptions for coffee break, phone calls). Real tasks can include "essential" interactions at a Web site (e.g., frequent, important tasks) as well as "unimportant" ones (e.g., surfing the Web). In contrast, representative tasks for lab-based testing, which include what are often called benchmark tasks, are typically approximations to essential tasks and are predefined by evaluators.

Marks in the second column of the table indicate methods (RUEMs 1, 2, and 3) for which remote evaluation occurs while users perform real on-the-job tasks and have no interaction with evaluators during task performance and/or evaluation. Marks in the third column indicate methods (RUEMs 4, 5 and 7) where users generally perform predefined representative (scripted) tasks, and users interact (briefly to significantly) with evaluators during the evaluation.

RUEM 6 is marked in the last column because this method does not involve users during evaluation but the evaluator might drive a usability inspection by simulating user performance of representative tasks. As the table shows, RUEM 2 is marked in two columns because, depending on the survey tool used to collect the data, users would perform either real or representative tasks. For example, users invited to participate in a Keynote® study perform predefined representative (scripted) tasks. With the Alertus® FormSurvey™, a survey can be displayed to users after

a specific action occurs (e.g., a visitor abandons a shopping cart).

Type of Data Gathered

Table 4 shows some types of usability data that are typically gathered using each remote evaluation method.

Table 4. Types of data collected via remote evaluation

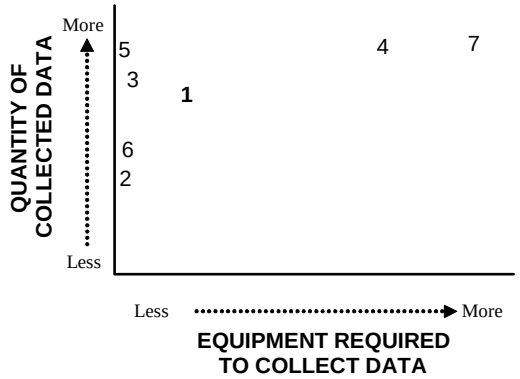
Remote Usability Evaluation Method	Types of qualitative data				
	Continuous video of entire session including audio with user comments	Evaluator notes of session, including critical incident data in the form of audio, text, and/or video clips	Log files, automatically recorded by system	Critical incident data identified, reported, and sent by users in the form of audio, text, and/or video clips	Subjective data from questionnaire
1. User-reported critical incident method				×	×
2. Remote questionnaire or survey					×
3. Instrumented or automated data collection			×		
4. Video-conferencing supported evaluation	×	×	X		×
5. Third-party lab-based usability testing*	×	×			×
6. Third-party usability inspection*		×			
7. Traditional lab-based usability testing	×	×			×

Several of the remote evaluation methods (1, 4, 5, and 7) have the capability to gather two or more types of qualitative data. The fact that a method can be used to gather a particular type of data does not mean that this kind of data is always gathered when using that method, or that it can be gathered at a reasonable cost. If a method shows a capability to gather a large number of types of data, it is likely that the equipment required for data gathering is more extensive and/or data analysis could be more complex (as discussed in the next section).

On the other hand, the ability to gather multiple types of data may be important to ensure that the most useful data (such as context information about critical incidents) are captured. For example, in the case of method 4, evaluators can obtain several different kinds of data such as an audio recording of user comments, a video of screen actions, evaluator notes taken during evaluation session, subjective information obtained from an online satisfaction questionnaire—and logs of user actions (e.g., keystrokes, visited pages) if using the Astoria Project. In contrast, method 3 yields only logs of user actions (e.g., click stream or visited paths on a site) and the remote questionnaire method yields only subjective opinions of users.

Quantity of Data Gathered and Equipment Used for Data Collection

In this and the next few sections, in a manner similar to that of Rouff [1996] and Williges [1984], we use graphs with unquantified axes to show our best estimates of relative standings of the methods with respect to measures such as "amount of analysis required", measures that are difficult to quantify absolutely.



Remote Usability Evaluation Methods:

1. User-reported critical incident method
2. Remote questionnaire or survey
3. Instrumented or automated data collection
4. Video-conferencing supported evaluation
5. Third-party lab-based usability testing
6. Third-party usability inspection

7. Traditional lab-based usability testing

Figure 4. Quantity of collected data and amount of equipment used for data collection

Figure 4 is a graph showing the relationship between "quantity of collected data", which is an intuitive concept of the volume of raw data collected and "equipment required to collect data", a quantity meant to distinguish, for example, methods requiring only a PC from those requiring a full usability laboratory with extensive video recording and editing devices.

As the Figure shows, RUEMs 5, 3, 6, and 2 have low equipment requirements for the evaluator: 5, because the third-party owns the necessary equipment; 3,

because logs of usage can be stored on a server; 6, because inspections typically do not involve audio or video recording; and 2, because questionnaires and responses can be stored on a server. In contrast, 4 has higher equipment requirements (such as a PC with the Astoria Project software and web cam), and 7 has the highest equipment requirements (such as video, audio and other laboratory equipment).

In terms of data collected, methods 5, 3, 4, 7, and 1 can yield large amounts, and methods 6 and 2 typically yield somewhat less raw data. The data collected using RUEM 2 varies depending on the specific technique used; for example, the data gathered by Keynote® produces more data than using an online survey with a small number of participants. The user-reported critical incident method produces lists of critical incidents, plus screen video data and textual comments and explanations from users. However, the matter of data quantity from this method (and other methods that produce a stream of output) is not so clear-cut, because the data come from on-going activity rather than a fixed number of sessions. Thus, more data result from more operation time and from more users reporting.

Relative Cost to Collect and Analyze Data

Figure 5 illustrates the relationship between the estimated cost to evaluators of collecting and analyzing data for each remote evaluation method. Costs (in all the figures) are limited to those incurred by an "in house" development team, mainly evaluators, and so include fees to third-party evaluators. However, this does not include costs that third-party evaluators incur on their own (which are presumably incorporated into the fees). These costs to the development team include equipment and technology amortization, time and effort (person-hours), external service fees (third party), training (including any developer and user training for evaluation), travel and time of user subjects, and travel and time for developers to visit user sites.

All quantities involving cost are given as relative costs. Although it would be useful to practitioners to have absolute costs for these methods, it is impossible to provide hard dollar values for these costs in this venue. Such costs will vary, depending on the details and circumstances of method usage, and could change rapidly, depending on changes in technology and markets. Where relative comparisons are likely to be more stable over time, absolute costs in a journal publication could end up being misleading.

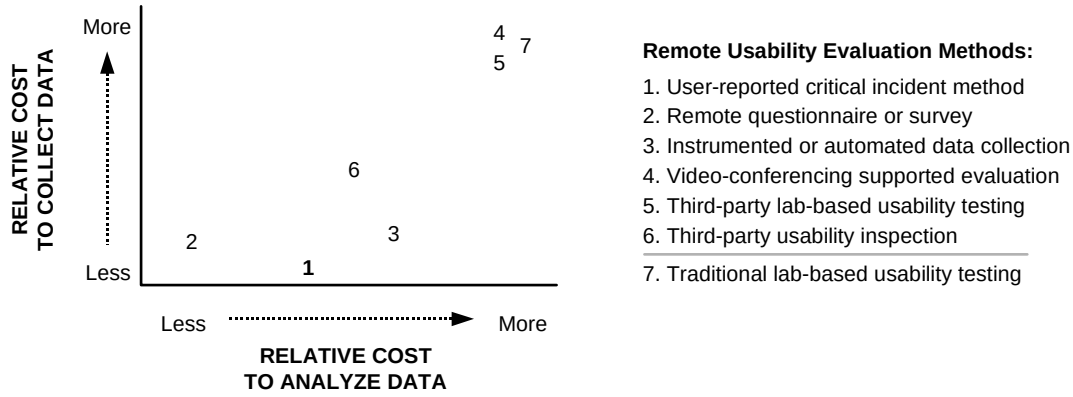


figure 5. Relative cost to collect and analyze data

As Figure 5 shows, RUEMs 1, 2, and 3 have a relatively low cost to collect data; 6, a medium cost; and 4, 5, and 7, a high cost.

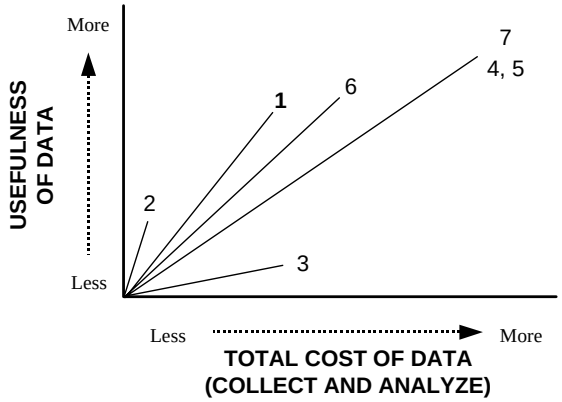
For RUEM 1 the cost for collecting data is almost entirely borne by users—users provide answers to questions and contextual data (screen video) associated with each critical incident. Analysis cost can be relatively low for RUEM 2 because the amount of data is often small. In RUEM 3, analysis cost is typically the cost to acquire and run automatic analysis software that looks through the data for patterns that might indicate usability problems, usually a computationally expensive process. RUEMs 4, 5, and 7 involve the highest cost to analyze data. Interestingly, RUEM 6 appears to make a good compromise when resources such as trained personnel are limited.

Usefulness of Data and Total Cost of Usability Data

Figure 6 shows a comparison of total data cost—collection and analysis—for a single project against an estimation of the usefulness of the collected data. "Usefulness of data" is an intuitive concept representing the quality or value of usability data in reaching the goal of formative usability evaluation: finding, analyzing (including diagnosing and understanding the problem and its causes), and fixing usability problems. It is a quantity used to distinguish, for example, lists of very specific critical incidents from voluminous clickstream data that is not good at indicating many kinds of specific usability problems.

Remote questionnaires (2) appear again in the lower left corner of the figure because costs to collect and analyze them are low and the value of the data is limited. This graph shows how instrumented methods (3) suffer from low usefulness of the data despite modest total cost. Once more, methods in the group

related to traditional lab-based evaluation (4, 5, and 7) appear in the upper right because they are the most costly but also yield the most useful data. The user-reported critical incident method (1) is somewhat lower on each axis than this group. It costs less than usability testing (5 and 7) or inspection (6) but produces data that are almost as useful.



Remote Usability Evaluation Methods:

1. User-reported critical incident method
2. Remote questionnaire or survey
3. Instrumented or automated data collection
4. Video-conferencing supported evaluation
5. Third-party lab-based usability testing
6. Third-party usability inspection

7. Traditional lab-based usability testing

Figure 6. Usefulness of data (for finding and fixing usability problems) and total cost of usability data

One interesting aspect of Figure 6 is the slope of the lines from the origin to the point representing each method. This slope is a measure of a kind of cost effectiveness or efficiency, namely usefulness per unit cost of data collected. Since the slope of a line in this graph represents a ratio of usefulness to cost, a higher slope suggests higher efficiency. Of course, as we have said, this graph is based on intuitive reasoning and represents only general characteristics of the methods. In specific instances of these methods as used by particular development groups, the slopes can vary.

The most efficient method according to the graph in Figure 6 is remote questionnaires and surveys (2) with a higher slope than the user-reported critical incident method (1). However, the high slope for this method comes from a quotient of low usefulness and low cost. This is a case that reflects the technical difference between efficiency and effectiveness. If the relative cost is low enough, efficiency can be very high, even though we get little in return. This limited usefulness alone may not be sufficient for the needs of the development team, regardless of the low cost.

However, if the budget is extremely limited, some data are better than none. But a questionnaire produces only subjective data and, as Elgin [1995] states, “Subjective feedback is generally harder to interpret

than objective feedback in a known setting…" More importantly, survey or questionnaire data cannot be the immediate and precise data about task performance that are essential for capturing the perishable details necessary to formative usability evaluation, as discussed in the early sections of this paper.

According to the graph of Figure 6, the user-reported critical incident method (1) has the next highest efficiency and has significantly more effectiveness (as indicated by the higher usefulness of data). It was, in fact, a goal during the development of this method [Hartson and Castillo, 1998] to produce an effective and efficient remote evaluation method, one that was less costly than traditional lab-based evaluation but that still maintained much of the usefulness in identifying usability problems.

Third-party usability inspection (6) comes next in terms of efficiency, perhaps as a good compromise possibility, especially in a shortage of trained usability engineering personnel. Because of the higher cost, remote methods (4 and 5), along with traditional lab-based testing (7), appear a bit less of a bargain, but they do yield the highest usefulness of data, an important consideration if results are more of a priority than cost.

Because the costs portrayed here are intended to be on a per-project basis, they include amortization of equipment, lab facilities, and training costs over several projects. Thus, the slopes of methods 4, 5, and 7 can vary depending on how these fixed costs are amortized. In particular, if the fixed costs of the laboratory set-up used in lab-based testing (7) are not well amortized (e.g., if the lab is not heavily used), third-party usability testing (5) or third-party usability inspection (6) might be more attractive than investing in a new lab-based facility (7) to do local testing or in equipment for teleconferencing (4). Additionally, both traditional lab-based usability testing (7) and third-party usability testing (5) can be quite expensive in terms of travel and time for representative users to visit the laboratory and could have decreased slopes due to this factor.

Finally, automated or instrumented data collection (3) fares the worst in our comparison. The cost can be high, due to both the investment in software design to do the complex analysis and the heavy computation that can be required to execute the programs for pattern matching. Most of the other methods, especially the critical incident method, search the "haystack" of user performance and behavior, sending to developers only the usability-related "needles". Instrumented or logging methods send large portions of the hay, the raw observational data, and the developers must find the needles using special software to analyze those data for patterns.

The usefulness of data yielded by software instrumentation is also limited because there are many kinds of usability problems that cannot be found by the method. As an analogy, a program for automatically analyzing text for writing quality is limited to finding spelling and grammar errors, but cannot assess more important writing issues such as word choices, semantics, and the logical sense of what is written and whether it conveys the ideas best to readers. Similarly, automatic analysis of usage logs can evaluate the more "shallow" mechanical factors such as the visited pages, visited links, areas of the page the user clicked on, etc. But it is much more difficult to determine whether a button label is effective in helping users predict the

functionality behind the button or which task structures are most natural to users.

Future Work

The work reported here has spawned future project possibilities in several areas including:

- A large scale survey among usability professionals about what RUEMs they use, for what situations they apply, and how useful they are compared to, for example, lab-based testing or another RUEM.
- Sample costs for each RUEM, for example, adding to the sample costs provided by Dumas and Hawley [2006] for video-conference supported remote evaluation.

References

[1] Abelow, Daniel (December, 1993). Automating Feedback on Software Product Use, CASE Trends, 15-17.

[2] Alertus® FormSurvey™ (2006). http://www.alertus.com/formsurvey.php [Visited: 07/01/2006]

[3] Castillo, J. C. (1997). The User-Reported Critical Incident Method for Remote Usability Evaluation. Unpublished Master's Thesis, Virginia Tech, Blacksburg, VA 24061 U.S.A.

[4] Dzida, W., Wiethoff, M., & Arnold, A. G. (1993). ERGOGuide: The Quality Assurance Guide to Ergonomic Software: Joint internal technical report of GMD (Germany) and Delft University of Technology (The Netherlands).

[5] Elgin, B. (1995). Subjective Usability Feedback from the Field over a Network. SIGCHI Bulletin, 27 (4), 43-44.

[6] ErgoLight® Usability Log Analyzer (2006) URL: http://www.ergolight-sw.com [Visited: 07/01/2006].

[7] Hammontree, M. J., Weiler, P., & Nayak, N. (1994). Remote Usability Testing. interactions, 1 (3), 21-25.

[8] Hartson, H.R., and Castillo, J. (1998). Remote Evaluation for Post-Deployment Usability Improvement. Proceedings of the Working Conference on Advanced Visual Interface (AVI'98)

[9] Hawley, M., and Dumas, J. (2006). Making Sense of Remote Usability Testing: Set-Up and Vendor Options. UPA 2006 Conference, Broomfield, CO.

[10] Hix, D., & Hartson, H. R. (1993). Developing user interfaces: Ensuring usability through product & process. New York: John Wiley & Sons, Inc.

[11] Keynote (2006) http://www.keynote.com [Visited: 07/01/2006].

[12] Microsoft® Windows NetMeeting™ (2006) URL: http://www.microsoft.com/windows/netmeeting/ [Visited: 07/01/2006]

[13] Raindance (2006) http://www.raindance.com/ [Visited: 07/01/2006]

[14] Siochi, A. C., & Ehrich, R. W. (1991). Computer Analysis of User Interfaces Based on Repetition in Transcripts of User Sessions. ACM Transactions on Information Systems, 9 (4), 309-335.

[15] Thomas, J. C., & Kellogg, W. A. (1989). Minimizing Ecological Gaps in Interface Design. IEEE Software, 6 (1), 78-86.

[16] TechSmith® Astoria Project (2006a). http://www.techsmith.com/astoria.asp [Visited: 07/01/2006]

[17] TechSmith® Camtasia Studio® (2006b). http://www.techsmith.com/camtasia.asp [Visited: 07/01/2006]

[18] TechSmith® Morae® (2006c). http://www.techsmith.com/morae.asp [Visited: 07/01/2006]

[19] Vividence™. (2002). Customer Experience Management: The Vividence Approach and Methodology (Technical Report) (Vividence is now Keynote).

[20] Webex (2006) http://www.webex.com/ [Visited: 07/01/2006]