Introduction

The Internet and its hosted applications remain a remarkably immature technology and application domain, given the exponential growth in websites and users. The combination of an immature technology, a distributed delivery and interaction mechanism, an immensely heterogeneous client base, and a usability-naïve Web developer community, has led to both predictable and unique usability problems, with an impact that far exceeds anything previously experienced.

Many researchers rate ease of use as being of critical importance to the E-Commerce process (Miles and Howes (2000), Tilson (1998), Vulkan (1999)). Bad usability is often blamed for causing the failure of sites (Helander and Khalid (2000), Rohn (1998)). Jahng et al. (Jahng (2000)) argues that the uncertainty and ambiguity of badly designed sites makes users feel uncomfortable, something that is unlikely to result in a sale.

Nielsen (2000) emphasises one significant reason for the importance of usability for the commercial Web. He refers to an inversion of the usability experience for the Web when compared to other products. Web users experience usability at a much earlier stage than conventional computer users and bad impressions will therefore potentially impact more severely on the company providing the site. Research has indicated that online customers will seldom stay around to investigate a site that puzzles them, or is difficult to learn to use. They tend to simply switch to another page in another site (Nielsen (2000), Nielsen and Norman (2000)). Bad usability could also reduce brand loyalty and lead to the relocation of buying preference.

The prevalence of low-usability websites is partly due to Web developers’ ease of access to the application domain resulting from the relatively minor barriers of entry. This leads to many usability-naïve and inexperienced developers operating in the domain. In this context various discount-engineering methods offer a solution. There are a number of approaches to validating the usability of websites. Three of the most popular are: empirical usability testing with a number of users, examination of Web logs and usability metrics. The first is expensive and probably not an option for most Web developers. The second is problematical for two reasons – it is essentially a post-mortem analysis and the results may come too late for remedial action to make a difference, and even if it is not too late, the Web logs are notoriously difficult to interpret (Nielsen and Norman (2000)). The final option is a low-resource (time, money, and skills), high-yield (usability improvement) option, and is thus suitable for use by inexperienced Web developers.

Metrics have been used successfully by Kirakowski and Claridge, who developed a commercial metric-based product called “The Web Analysis and MeasureMent Inventory” (WAMMI) usability questionnaire, which originated from a tool for evaluating desktop applications [W8].
WAMMI, although Web-specific, is not tuned for E-Commerce site evaluation and does not consider the uniqueness of the E-Commerce task. Ivory et al. (2001) have worked on quantifying the measurement of information-centric Web pages in terms of presentation. They developed a tool called TANGO (Tool for Assessing Navigations and Information Organisation) (Ivory, Sinha and Hearst (2000)), which was used to evaluate over 400 websites. Their tool is highly automated because it evaluates simple page elements rather than semantic content and context, both of which are addressed by the metrics introduced in this chapter.

The rest of this chapter discusses a combination of a task-based and metric-based approach to enhancing Web usability. This is a viable yet novel technique for providing inexperienced developers with a tool that can be used to improve the quality of their sites.

Why Focus on E-Commerce Tasks?

The Internet is the largest information source ever created. Users of the Internet are said to surf the net—a euphemism for browsing and searching for sites of interest. Indications are that initially observed rather aimless behaviour is changing and maturing: Studies by the Radio and Television News Directors Foundation and the Pew Research Center show that people want to accomplish something online. They're not aimless "surfers" anymore looking for a novel site or anything of passing interest. Instead, the average Net user seems to be becoming more goal-oriented and interested in finding specific information and communicating with others [W4].

Belew (2000) examines the Web browsing and searching process and finds that three steps are involved: asking a question, constructing an answer, and assessing the answer. While the usability of solely information-specific sites is important, the fact that the usability of E-Commerce sites could impact negatively on a company’s regular retail trade as well as their E-Commerce trade makes it vital for designers to produce usable E-Commerce sites [W1]. It can be argued that for these strongly goal-directed applications the bulk of a system's usability is accounted for by task focus. Travis [W6] recommends:

‘Allocate your design activity to reflect the needs of end-users, and put the bulk of your effort into task focus. Users will thank you because they will be able to work more effectively. You know a system has task focus when you get a warm feeling that the person who designed this system knew what they were doing. You find you are able to use the system to do exactly what you want.’

It is important to note Diaper and Addison's (1991) claim that a task will be strongly dependent on the user's mental model of the computer and (for this study, on-line), documentation. Task decomposition (perhaps too mechanistic), knowledge based techniques and entity-relationship based analysis (now replaced by UML), are three different but overlapping approaches to task analysis (Dix et al. (1998)) and of these, the second is especially relevant to E-Commerce. Knowledge-based descriptions of tasks are important because they identify:

1. The plan for carrying out the task,
2. The knowledge or concepts required, and
3. The interaction between different kinds of knowledge.

The designer therefore gains an understanding of the commonalities that exist between tasks in terms of knowledge requirements and anticipated plans for execution (Johnson et al. (1984)). In this regard Brinck et al. (2002) further note that “task analysis is critical to providing a system that is efficient to use and easy to learn while not exceeding human limitations”. In addition, the
high-level goals specified in the task analysis make explicit the functionality that is built into the system. Thus there is little confusion about the intended purpose of the site. Because it makes explicit the procedural knowledge expected from the user, it clarifies learning requirements and can provide the basis for training materials.

It can be expected that a task analysis should yield, inter-alia, the following benefits when applied to the E-Commerce interaction session (Maguire (1997)):

1. It provides knowledge of the tasks that the user wishes to perform i.e. find, select and purchase.
2. It is a reference against which the value of the system functions and features can be tested.
3. It is a cost-saving exercise because failure to allocate sufficient resources to the task analysis activity increases the potential for costly problems arising in later phases of development.
4. It makes it possible to design and allocate tasks appropriately and efficiently within the new system by means of the use of task analysis.
5. It enables us to specify the functions to be included within the system and the user interface more accurately.

Most users of E-Commerce systems will not have been trained in their use. The user interface will therefore have to be designed with great care so that the user can discover everything that is task-supportive from the system, based on the perceptible system state. The designer of the user interface must be sure to bestow rational behaviour on the application – ensuring that the application behaves in a way that is reasonable and intelligible. By concentrating on the E-Commerce task the developer can move closer to a system that the user can use intuitively.

Substandard E-commerce sites currently focus on available technology rather than the user’s task. By shifting the focus from technology to customers the usability benefits will be inter alia [W6]:
1. An improved conversion rate (lookers/bookers ratio) and hence improved sales. On a high-volume E-commerce site, raising the conversion rate by one tenth of 1% can add thousands in revenue per month.
2. Increased hits. A satisfied customer provides free word-of-mouth exposure — bringing more customers to the site.
3. Avoiding leakage to competitors. One bad customer experience could cause a customer to abandon a site permanently and will possibly result in bad publicity
4. Focusing developers on important business metrics which will result from being clearly responsive to customer needs. These include conversion rates; revenue per order; acquisition cost per new customer; percentage repeat buyers; percentage revenue from repeat buyers; abandoned shopping carts.
5. Reducing support staff costs by allowing customers to carry out maintenance tasks online.
6. Helping developers to stay focused on customers’ tasks and goals, rather than on the Web as a technology tool
7. Highlighting wastage of development time and establishing guidelines for the next move.

The next section will identify two distinct phases of the purchase task of the E-Commerce shopping experience, and the following chapter will analyse the E-Commerce task with these phases in mind.

Two Phases of the E-Commerce Task

Some work has been done in analysing user behaviour with respect to E-Commerce sites. Guttman et al. (1998) identify six stages of customer purchasing behaviour: need identification,
product brokering, merchant brokering, negotiation, purchase and delivery, and service and evaluation. O'Keefe and McEachern (1998) propose a model with only five processes: need recognition, information search, evaluation, purchase, and after-purchase evaluation. Singh et al. (1999) break up the E-Commerce process into three activities: identifying and finding a vendor, purchasing and tracking. We will examine only one of Singh's processes – namely the one that everyone refers to as the purchase task. This task can be split up into two distinct phases, as shown in Figure 17.1:

Figure 17.1. The Two Phases and Ten Stages of the Purchase Task. (Renaud et al., (2001))

1. Look, See and Decide (LSD): This stage will typically be used to look at available products, compare them, and to make a decision about whether or not to purchase products. This may be done one or more times until the consumer has found products that satisfy his or her needs. This phase is intensely user-driven because the user is looking at and assimilating information continuously. It has the following substages which can be traversed iteratively and in varying sequences:

   Welcome; Search; Browse; Choose.

2. Checkout: When users trigger this stage they have made their choice of offered products and have decided to make a purchase. They now have to provide certain details, such as their address and credit card details. This stage is system-driven and changes the paradigm of the interaction process from user initiative to system initiative. Feedback is of critical importance during this stage. Users who feel that they have lost control can simply leave the site without any embarrassment, unlike a user who is standing at a checkout till in a supermarket. This stage is typically composed of at least the following steps, which should be navigated in a logically serial fashion:


Some Websites will have all these stages integrated into one page (e.g. www.amazon.co.uk) but the implied functionality is the same – each of these categories of information must be provided so that the transaction can be carried out. Brinck, Gergle and Wood (2002), combine UML use case analysis with hierarchical task analysis into a powerful technique. They identify two use cases for a book purchasing scenario namely “Buy Book” and “Complete Order”, which coincides with the LSD and Checkout phases as identified above.
The user drives Belew’s searching and browsing process during the LSD phase, asking question after question until a particular item has been located. The process is somewhat different during the Checkout phase, because the system drives the process — asking the user questions, receiving and assessing the answers. Most Web-design guidelines do not take these diametrically opposing operating paradigms into account, even though the principle of dialog initiative and system versus user pre-emption is well established (Dix et al. (1998)).

**Preliminary Task Analysis**

A simple computer-operating model may serve as an effective basis for an understanding of the goal-directed nature of an E-Commerce task execution. This model can also serve to further highlight the tasking difference between the two phases of the E-Commerce shopping process. A definition for task analysis that is suitable within the context of this application domain is that offered by Dix et al. (1998) – they describe task analysis as the identification and description of the interactive system user’s problem space, in terms of domain, goals, intentions, and tasks.

**Figure 17.2:** Task Analysis based on a Simple Iterative Computer Operating Model (adapted from Woodson et al. (1992))

The focus of this part of the study will be on identifying differences between the nature of the task during the LSD (which includes search-and-browsing, selection and the shopping cart), and the Checkout phases of the E-Commerce purchase interaction process.

The nature of the shopping task differs significantly during the LSD (Look-See-Decide) and Checkout phases. The LSD phase is, in essence, a user-driven iterative browsing and selection task with (possibly) less well-defined goals and a larger number of possible actions. The Checkout phase is a system-driven pre-defined, linear task with well-defined goals and sub-goals, and with a smaller number of predefined actions.

During the LSD phase there will be three types of goals:
• browsing (searching), for the shopping object,
• categorizing (searching-result), the shopping objects, and
• specifying (selecting) a shopping object for the shopping cart.

The nature of the interaction is such that the customer should be kept interested in the results of the search-type goals, thus retaining them on-site – the focus is on discouraging user dropouts through abandonment of the goal or by linking them off-site (Dix (1999)). System errors and poor response times during this phase are perceived to be less serious by the user (but not by the Website owner), since they may result in shopper abandonment.

The checkout phase has a single goal – completing the financial transaction (as defined by the contents of their shopping cart and their shipping preferences) as quickly and securely as possible with the minimum of disruption and roadblocks [W1]. Accordingly this phase has a set of linear and intentionally fairly rigid sub-goals. The focus here is not on user entertainment but on completing the transaction rapidly and securely – before shoppers change their minds about their shopping carts and the related cost [W1]. This implies that response times, and clear feedback on reasons for delays, are more important here than during the LSD phase. Because errors could have a more serious (security and financial), impact, a well-designed user help function and clear explanatory sub-system are required. It is also easy to provide the user with obvious and intuitive navigation clues as to where they are in the process by using progress or stage indicators. The trend should be to strive for the minimum number of pages or stages – rather have the user scrolling moderately than clicking through to a larger number of small pages [W1]. This is in stark contrast to having as much as possible of the relevant information immediately visible in the LSD stage (Veen (2001)). Simplifying this process will ensure that there will be a smaller incidence of user dropout and shopping cart abandonment during this phase – provided additional costs such as shipping are shown as soon as possible. The effect of an error at this stage will affect the sequence of the sub-goals, and will make this phase non-linear (i.e. 'loopy').

When these aspects are applied to the model as presented in Figure 17.2 the following should be noted for the two phases:

1. More effort will be required for system activation (S1) in the LSD phase when compared to the checkout phase – for example, the customer has to have an established Internet session. The customer also needs to know about the site. This part is well understood by marketing professionals and sites are often well advertised in the media. Unfortunately this level of attention is often not paid to other aspects of the E-Commerce experience.
2. Goal formulation (S2) may be less clear in the LSD phase as compared to the checkout phase – the customer may want to re-evaluate options and re-formulate goals based on the range, price, and availability of the shopping objects during the outcome of a set of search results.
3. The intermediate stages (S2 – S6) are less proscribed for the LSD phase, and there will be a natural tendency to loop back to S2 during this phase – for example, if the response (at S5) takes too long.
4. The S3 stage is often trickier for the user to formulate in the LSD phase. The user may have some vague idea of an item he or she needs, but may have difficulty formulating a query. For example, the user may have heard about a popular autobiography by an Irish teacher who grew up in Limerick. The user types in many different search criteria – “Irish”, “teacher”, “Limerick” before perhaps finding the book *Angela's Ashes* by Frank McCourt by browsing through the list of available autobiographies.
5. Interpretation of the response (S6) will be more difficult during the LSD phase when compared to the checkout phase – the customer may be presented with a range of shopping objects from which to choose compared to the linear progression during the checkout phase.

6. Measuring the success of the task at S5 will be more difficult for the LSD phase – the customer is dealing with a goal achievement based on an electronic description rather than confirmation of a familiar financial transaction as in the checkout phase.

7. The result of an error (which may be at S7 $\rightarrow$ S1, or S4 $\rightarrow$ S2) will be deemed to be less serious during the LSD phase than during the completion of a transaction in the checkout phase.

8. The transition S7 $\rightarrow$ S2 may be traversed during the LSD phase without an error having been made – it could happen as a result of a reformulated goal.

9. The transition S7 $\rightarrow$ S8 could be made as a result of abandonment even though the goal has not been achieved – since the site may not stock the required object.

10. The activity distance S2 $\rightarrow$ S7 should be as short as possible for the checkout phase with achievement of S7 always clearly visible, perhaps by means of stage indicators.

11. The S3 $\rightarrow$ S9 and S5 $\rightarrow$ S9 loops should be minimised by ensuring good site usability.

It is necessary to translate the discussion of this section into some set of recommendations so that developers have guidelines to follow in order to ensure that adequate feedback is provided. Veen (2001) refers to the difficulty of evaluating websites. Developers using traditional usability-testing methods are often faced with an iterative and time-consuming evaluation process involving a number of users – until every perceivable problem is solved. The provision of a set of easily applied metrics should make it easier for E-Commerce site developers to profit from accumulated research results and make the evaluation process a little less daunting.

**Task-Weighted Evaluation Metrics**

The previous section discussed the differences between the two different phases of the E-Commerce task. It is fitting for the two phases to have different evaluation metrics as well – as befitting their different paradigms and needs – and for these metrics to be weighted according to their impact on usability.

We previously reported on a set of suitable evaluation metrics for measuring feedback-related usability of three E-Commerce websites (Renaud et al. (2001)). The evaluation criteria used were equally weighted. In certain cases, however, it may be advantageous to prioritize some of the criteria by means of a selective, unequal, weighting. Examples of how to do this may be found in Levi and Conrad (1996). They describe the application of Nielsen and Mack's (Nielsen (1994)) usability guidelines to the evaluation of a set of Web pages. After the evaluation they modify the list based on feedback from their two different (HCI and Web developers) evaluation teams, and produce a new list by assigning severity ratings to each usability violation found on a five-point scale. In addition they also prioritize on the basis of the frequency of occurrence of the usability problem. Their scale varies from 0=Not a usability problem; 1=Cosmetic; 2=Minor; 3=Major; to 4=Catastrophic problem. They produce a list of usability violations, which contains both frequency and severity information. Nielsen (2001) later produced a guideline for severity ratings based on the same scale, but then expanded on this by noting that the severity of a usability problem was a combination of four factors: The frequency with which the problem occurs, the impact of the problem on users if it occurs, the persistence of the problem, and the market (product popularity) impact of the problem [W5].
Along the same lines, Bastien and Scapin (1992) refer to the amount and importance of usability problems found. Another technique applies a strength of evidence scale to a set of evaluation criteria [W2]. These criteria are based on the type and number of research experiments that may support, or discount, the specific criterion. W3.org [W3] prioritizes in terms of (accessibility) guidelines that must be applied, should be applied, or may be applied.

One suitable weighting approach, which we believe to be in part novel, is to assign a task weighting to each of these previously equal-weighted criteria scores. This task weighting has two components:

1. A task repetition component (R). The task repetition component is an indicator of how often this task or activity will be encountered during the interaction. A weight of 0.1 will indicate it to be of low occurrence – implying that it only happens in exceptional cases, whereas a factor of 0.2 means that it happens very seldom, and a weight factor of 1.0 will indicate that this type of activity occurs regularly during the interaction. A value of 0 indicates absence of the activity – indicating that it should not play a role in the calculation of the overall score.

2. A task complexity component (C). The task complexity component reflects the inherent degree of difficulty in executing the task or activity. A weight factor of 1.0 indicates that the activity is highly complex, requiring extensive background and operational knowledge, or requiring a high degree of complex interaction. A weight factor of 0.1 indicates that the task is simple, with low interactivity. A weight value of zero is not possible because it indicates that no interaction is required.

Each of these two components either amplify or attenuate the contribution of a specific usability feedback criterion to the overall score. The overall value, should be a more faithful reflection of the website's overall usability than that rendered by unweighted metrics.

In support of our view it should be noted that Brinck et al. (2002) distinguish between the frequency and priority of tasks and that they (correctly) note that the starting point for their HTA (Hierarchical Task Analysis), should be important tasks that occur frequently. Similarly Nielsen’s (2001) ‘impact and persistence of the problem’ could make the problem context a high-priority task. Lastly, Bohmann has developed and tested task metrics for a quantitative usability evaluation. These metrics make it possible to calculate the usability effect of redesign efforts. The two main metrics are: (1) Task Time – time to complete a task or set of tasks and (2) Task Errors – number of errors per task [W7].

The following section will evaluate four E-Commerce book sellers’ websites using this technique.
Evaluation of E-Commerce Sites

It is difficult for developers, given a list of guidelines, to know which to follow. For example, developers are told to have the most important information visible to the user without scrolling. They are also told to provide the user with enough information to keep them interested in the site thus increasing the chance of a sale. Which of these is more important? We therefore propose the use of a set of metrics that can be used by developers to evaluate each page of a Website. Additionally we propose an approach of metric weighting that involves the use of the task complexity and task repetition components as discussed previously.

The following section will describe how these metrics were applied to a selection of four E-Commerce sites, and comment about the efficacy of the proposed evaluation mechanism. In order to evaluate E-Commerce Web pages, a raw score is given for each of the questions (metrics) as follows (Ravden and Johnson (1989)):

<table>
<thead>
<tr>
<th>Score</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Never (Never available)</td>
</tr>
<tr>
<td>1</td>
<td>Sometimes (Rarely available)</td>
</tr>
<tr>
<td>2</td>
<td>Mostly (Usually available)</td>
</tr>
<tr>
<td>3</td>
<td>Always (Always available)</td>
</tr>
</tbody>
</table>

**List 17.1:** The Four Point Categorical Rating Scale for the Usability Metrics

The first step in the evaluation is the raw scoring of the usability metrics. The scores are determined per E-Commerce site, per phase (LSD and Checkout), per stage within the phase, and also per page, as a ratio to the maximum score. The scores for each metric in each stage are calculated by adding up the score for each page making up the stage and awarding a total for each particular metric feature. The scores for each feature are then totaled to arrive at a percentage per site per purchasing stage to arrive at a raw score. It is important to note that the evaluator should not feel constrained by the list of metrics given here – these were adapted and selected from a much larger list (11 sections and 179 metrics) developed and extensively tested by Ravden et al. (1989). It is likely that differences in E-Commerce sites may require the evaluator to re-visit this more comprehensive list and add to (or subtract from) the list of metrics given here. It is also recommended that more than one evaluation (and evaluator) be used to arrive at the raw metric scores – three data sets can be considered to be the absolute minimum. The individual scores from the data sets should be averaged as an input to the second step.

The second step assigns values for the two task weight components (R+C), based on the evaluators’ experiences with the site during the metric scoring step. These values are designed be have little effect initially on the raw scores until the evaluator develops more confidence in applying the correction. The natural tendency will be to choose median values of close to 0.5 for both the task repetition and complexity values which will imply that the weighting adjustment will effectively be 1, \((R+C) = (0.5+0.5)\). Thus initially the adjustment will be no worse than the unadjusted raw metric scores. Ultimately as experience is gained in the use of the weight factors, the weight adjustments could realistically have a large influence on the metric score – consider a low complexity, low repetition value of 0.5 compared to a high repetition high complexity value of 1.5. The metric adjusted by the first would only contribute one third as much to the overall usability score for the E-commerce site when compared to the second metric.

A third step would then be to eliminate those metrics with particularly low \((R+C)\) values (for example if \((R+C)<0.6\)), from the evaluation – this would partly alleviate the problem of a tendency towards an average of 1 for all \((R+C)\) values when a large number of metrics are used.
To demonstrate our approach, consider row number 9 in Table 17.4 with the question: "Is it clear what the user has to do to complete the task?" In order to score the new Amazon.com site do the following:

**Step 1:** Each reviewer does a dry run through the checkout phase of Amazon.com and records all actions and responses for each page.

**Step 2:**

a) For each page, assign a score for the particular usability metric. For example, the first page in the Amazon.com checkout process displays a login script requiring the user to provide an email address and password. A score of 3 is assigned to this page since it is obvious that the user needs to enter the details and click on the "Sign In" button. The full 3 points are given since the user does not need to scroll down to find the button and no other interpretation is possible. If the user has to search for the button or if two buttons appear with similar functions the score may be reduced.

b) Add the scores for each page within the stage together. Now average the scores assigned by the participating reviewers to arrive at an assigned score (referred to as Score in the table). In this case we arrived at a value of 7 for that particular metric. The maximum score is 9 since this stage consists of 3 pages, each of which has a maximum score of 3 (See List 17.1). This is referred to as MaxScore in the table.

**Step 3:**

a) Based on the evaluators’ observations during steps 1 and 2, assign suitable values to the task repetition and complexity weighting for that specific metric. In this case we used a value of 0.5 for the repetition since the actions to be taken in these pages are neither particularly frequent nor infrequent and therefore an average value of 0.5 has been assigned. The complexity of this task was assigned a value of 0.7 indicating that this task falls between an average and a highly complex task. This is because the order review page requires the use of extensive external knowledge and interpretation in order to carry out the task correctly.

b) The value of 16 is the total of all the R and C values for any specific phase in a site. Each R+C value represents a fraction of the total usability environment for the site. Comparing the R+C values for particular metric with the total value is an indication of how important that specific metric is to the total site usability (referred to as TotalR+C).

c) To arrive at the final usability coefficient (UC) for the metric we apply this formula:

$$UC = \frac{Score}{MaxScore} \times \frac{(R+C)}{TotalR+C}$$

To complete the table we need to calculate an overall usability score for each phase per site in order to facilitate a comparison between sites. The % usability score based on raw (non-task-weighted) scores is:

$$Raw = \frac{\sum Score}{\sum MaxScore} \times 100$$

The percentage usability, based on task-weighted scores, is calculated as:

$$Task-weighted = \frac{\sum UC}{\sum UC} \times 100$$

The normalised ratios make it easier to compare usability scores and are expressed as ratios relative to the highest scoring site. For sites other than the top site the score is calculated as follows:
Task-weighted score / Top-Site’s Task-weighted score

A suitable list of evaluation metrics as derived from previously reported results (Renaud et al. (2001)) is shown in Tables 17.1 and 17.2 for the LSD and Checkout phases respectively. Typical task weighting factors for each metric are as indicated. To use these tables the original (raw) score for each metric is multiplied by the factor given in the table for each metric. Of particular interest in these two tables, would be criteria with associated task or activities that have either high combined (i.e. repetition and complexity (R+C)), weight factors, or very low combined weight factors. This could imply that these task components are proportionally either more, or less important to the usability evaluation.

Table 17.1. User task metrics for the LSD phase.

<table>
<thead>
<tr>
<th>LSD Phase: Metrics for the evaluation of User Task support</th>
<th>Task Weighting Factors (Repetition + Complexity)</th>
</tr>
</thead>
<tbody>
<tr>
<td>S3 Is it clear how the user must search for a product?</td>
<td>(0.8+0.5)/8.5 = 15% (&gt;1)</td>
</tr>
<tr>
<td>S3 Are different types of information clearly separated?</td>
<td>(0.8+0.9)/8.5 = 20% (&gt;1)</td>
</tr>
<tr>
<td>S4 Is it clear what needs to be done to select a product?</td>
<td>(0.1+0.6)/8.5 = 8% (&lt;1)</td>
</tr>
<tr>
<td>S5→S9 Does the system inform the user of reasons for delays?</td>
<td>(0.8+0.5)/8.5 = 15% (&gt;1)</td>
</tr>
<tr>
<td>S7→S2 Does the search engine offer alternatives if the search fails?</td>
<td>(0.5+0.8)/8.5 = 15% (&gt;1)</td>
</tr>
<tr>
<td>S2→S3 Can the user undo a product selection?</td>
<td>(0.1+0.5)/8.5 = 7% (&lt;1)</td>
</tr>
<tr>
<td>S2→S3 Does the system allow the user to explicitly check on previous searches?</td>
<td>(0.5+0.5)/8.5 = 12% (1)</td>
</tr>
<tr>
<td>S8→S9 Is it clear how the transition to checkout can be made?</td>
<td>(0.1+0.5)/8.5 = 7% (&lt;1)</td>
</tr>
<tr>
<td>Average Task Weighting Factor for this Phase:</td>
<td>8.5/8.0 = 1.06</td>
</tr>
</tbody>
</table>

Notes (Table 17.1): For example (0.8+0.5)/8.5 = 15% (>1): (R+C) = (Task Repetition Value + Task Complexity Value)
8.5: Total of the task (R+C) values for the specific set of usability metrics.
15%: Percentage portion (contribution) to the total task weight.
(>1): (R+C>1) Implies an increased weighting for any usability score for this metric.

Table 17.2. User task metrics for the checkout phase.

<table>
<thead>
<tr>
<th>Checkout Phase: Metrics for the evaluation of User Task support</th>
<th>Task Weighting Factors (Repetition + Complexity)</th>
</tr>
</thead>
<tbody>
<tr>
<td>S2,S3 Are possible actions clear?</td>
<td>(0.8+0.7)/16 = 9% (&gt;1)</td>
</tr>
<tr>
<td>S3 Are instructions and messages concise, clear and unambiguous?</td>
<td>(1+0.8)/16 = 11% (&gt;1)</td>
</tr>
<tr>
<td>S3,S5→S9 Can the user easily back out of the process?</td>
<td>(0.1+0.8)/16 = 6% (&lt;1)</td>
</tr>
<tr>
<td>S4 Is the required format of user actions clearly indicated?</td>
<td>(0.5+0.8)/16 = 8% (&lt;1)</td>
</tr>
<tr>
<td>S5→S9 Does the system inform the user of the reasons for delays?</td>
<td>(1+0.5)/16 = 9% (&gt;1)</td>
</tr>
<tr>
<td>S6 Are user actions linked to changes in the interface?</td>
<td>(0.8+0.5)/16 = 8% (&lt;1)</td>
</tr>
<tr>
<td>S6 Is there always an appropriate response to user actions?</td>
<td>(0.8+0.8)/16 =10% (&gt;1)</td>
</tr>
<tr>
<td>S6 Does the user explicitly confirm the final purchase?</td>
<td>(0.1+0.4)/16 = 3% (&lt;1)</td>
</tr>
<tr>
<td>S6 Does the system indicate the current stage?</td>
<td>(0.3+0.5)/16 = 5% (&lt;1)</td>
</tr>
<tr>
<td>S6 Can users check on inputs provided during the process?</td>
<td>(0.1+0.5)/16 = 4% (&lt;1)</td>
</tr>
<tr>
<td>S7→S8 or S7→S2/S3/S4 Does the system inform the user of the success or failure of their actions?</td>
<td>(1+0.5)/16 = 9% (&gt;1)</td>
</tr>
<tr>
<td>S7→S2 Do error messages indicate the what, where and why and how to recover?</td>
<td>(0.5+1)/16 = 9% (&gt;1)</td>
</tr>
<tr>
<td>S7→S8 Is it clear what the user must do to complete the task?</td>
<td>(0.5+0.7)/16 = 8% (&gt;1)</td>
</tr>
<tr>
<td>Average Task Weighting Factor for this Phase:</td>
<td>16/13 = 1.23</td>
</tr>
</tbody>
</table>

Table 17.3. Results for applying the user task metrics for the LSD phase

<table>
<thead>
<tr>
<th>Evaluation of User Task Support:</th>
<th>Amazon (New)</th>
<th>Amazon (Old)</th>
<th>Kalahari</th>
<th>BOL</th>
</tr>
</thead>
<tbody>
<tr>
<td>LSD Stage</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1. Is it clear what a user must do to search for a product?</td>
<td>6/9(0.8+0.5)/8.5</td>
<td>6/9(0.8+0.5)/8.5</td>
<td>6/9(0.8+0.5)/8.5</td>
<td>7/9(0.8+0.5)/8.5</td>
</tr>
<tr>
<td>2. Does the search engine offer alternatives if a search fails?</td>
<td>9/9(0.5+0.6)/8.5</td>
<td>9/9(0.5+0.5)/8.5</td>
<td>3/9(0.5+0.5)/8.5</td>
<td>0/9(0.5+0.5)/8.5</td>
</tr>
<tr>
<td>3. Does the system inform the user of the reasons for delays?</td>
<td>5/9(0.8+0.5)/8.5</td>
<td>5/9(0.5+0.5)/8.5</td>
<td>3/9(0.5+0.5)/8.5</td>
<td>3/9(0.5+0.5)/8.5</td>
</tr>
<tr>
<td>4. Are different types of information clearly separated?</td>
<td>8/9(0.8+0.9)/8.5</td>
<td>7/9(0.8+0.8)/8.5</td>
<td>6/9(0.8+0.8)/8.5</td>
<td>9/9(0.8+0.8)/8.5</td>
</tr>
<tr>
<td>5. Is it clear what needs to be done to select a product?</td>
<td>9/9(1+0.6)/8.5</td>
<td>9/9(1+0.5)/8.5</td>
<td>6/9(0.1+0.5)/8.5</td>
<td>6/9(0.1+0.5)/8.5</td>
</tr>
<tr>
<td>6. Can the user undo a product selection?</td>
<td>9/9(0.1+0.5)/8.5</td>
<td>9/9(0.1+0.5)/8.5</td>
<td>6/9(0.1+0.5)/8.5</td>
<td>8/9(0.1+0.5)/8.5</td>
</tr>
<tr>
<td>7. Is it clear what must be done to make the transition to Checkout?</td>
<td>7/9(1+0.5)/8.5</td>
<td>6/9(1+0.5)/8.5</td>
<td>0/9(0.1+0.5)/8.5</td>
<td>9/9(0.1+0.5)/8.5</td>
</tr>
<tr>
<td>8. Does the system allow users to explicitly check on previous searches?</td>
<td>8/9(1+0.5)/8.5</td>
<td>8/9(1+0.5)/8.5</td>
<td>8/9(1+0.5)/8.5</td>
<td>8/9(1+0.5)/8.5</td>
</tr>
<tr>
<td>Percentage: Raw</td>
<td>59/72 = 81.9%</td>
<td>59/72 = 81.9%</td>
<td>59/72 = 81.9%</td>
<td>59/72 = 81.9%</td>
</tr>
<tr>
<td>Task-weighted</td>
<td>8/9(0.8+0.9)/8.5</td>
<td>7/9(0.8+0.8)/8.5</td>
<td>6/9(0.8+0.8)/8.5</td>
<td>9/9(0.8+0.8)/8.5</td>
</tr>
<tr>
<td>Normalized Ratio: Raw</td>
<td>1.0</td>
<td>0.865</td>
<td>0.509</td>
<td>0.712</td>
</tr>
<tr>
<td>Task-weighted</td>
<td>1.0</td>
<td>0.843</td>
<td>0.647</td>
<td>0.714</td>
</tr>
</tbody>
</table>

The results from Table 17.3 show that:
1. Applying the task weighting has decreased the overall usability difference between the best site – Amazon (2002), and the worst site – Kalahari. Small changes were observed for BOL. Elimination of low value (R+C) task metrics will result in larger differences between weighted and un-weighted results.
2. For the lowest usability site (Kalahari), applying the task weighting results in a significant increase in its overall usability score. This would imply that Kalahari does focus on better usability for important tasks compared to other metrics which have lower (R+C) values.
3. Applying the weighting factors has emphasised the usability differences between the new and old Amazon sites. The new Amazon has improved considerably on its usability score for the LSD phase – this is mainly due to higher scores for content layout, information presentation, the provision of a history function, and more obvious navigation to next stages in the book purchase task.
4. High (R+C) criteria include activities associated with the presentation of information, and instruction-oriented actions.
5. Low (R+C) criteria include product selection and de-selection actions, and undo facilities.
6. The raw scores also yield useful information by themselves. They provide an evaluation mechanism which can be used by developers to flag problem feedback areas. For example the old Amazon website did not have a search history facility and as a result scored 0 for metric 8. This was corrected in the new Amazon site.

Table 17.4. Results for applying the user task metrics for the checkout phase

<table>
<thead>
<tr>
<th>Evaluation of User Task Support:</th>
<th>Amazon – New (3 stages)</th>
<th>Amazon – Old (6 stages)</th>
<th>Kalahari (3 stages)</th>
<th>BOL (5 stages)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Are instructions and messages concise, clear and unambiguous?</td>
<td>8/9(1+0.8)/16 = 0.100</td>
<td>12/18(1+0.8)/16.5 = 0.073</td>
<td>6/9(1+0.8)/16.5 = 0.073</td>
<td>11/15(1+0.8)/16.5 = 0.080</td>
</tr>
<tr>
<td>2. Are possible actions clear?</td>
<td>7/9(0.8+0.7)/16 = 0.073</td>
<td>12/18(0.8+0.8)/16.5 = 0.073</td>
<td>5/9(0.8+0.8)/16.5 = 0.054</td>
<td>12/15(0.8+0.8)/16.5 = 0.078</td>
</tr>
<tr>
<td>3. Is the required format of user inputs clearly indicated?</td>
<td>8/9(0.5+0.6)/16 = 0.072</td>
<td>15/18(0.8+0.5)/16.5 = 0.076</td>
<td>7/9(0.5+1)/16.5 = 0.071</td>
<td>10/15(0.5+1)/16.5 = 0.061</td>
</tr>
<tr>
<td>4. Are user actions linked to changes in the interface?</td>
<td>7/9(0.8+0.5)/16 = 0.063</td>
<td>13/18(0.8+0.5)/16.5 = 0.057</td>
<td>6/9(0.8+0.5)/16.5 = 0.053</td>
<td>12/15(0.8+0.5)/16.5 = 0.063</td>
</tr>
<tr>
<td>5. Is there always an appropriate response to user actions?</td>
<td>6/9(0.8+0.8)/16 = 0.067</td>
<td>12/18(0.8+0.8)/16.5 = 0.065</td>
<td>6/9(0.8+0.8)/16.5 = 0.065</td>
<td>13/15(0.8+0.8)/16.5 = 0.084</td>
</tr>
<tr>
<td>6. Does the system inform the user of the success or failure of their actions?</td>
<td>8/9(1+0.5)/16 = 0.083</td>
<td>14/18(1+0.5)/16.5 = 0.071</td>
<td>6/9(1+0.5)/16.5 = 0.061</td>
<td>11/15(1+0.5)/16.5 = 0.067</td>
</tr>
</tbody>
</table>
The results from Table 17.4 show that:

1. The task weighting has improved the score of BOL, but decreased that for all three other sites. The smaller changes when compared to Table 17.3 are in part due to the larger number of metrics used in this table when compared to Table 17.3 – i.e. there is to some extent an averaging around the mean of the (R+C) value. This could be avoided by eliminating all metrics with low (R+C) values (for example values < 0.6) from the scoring.

2. The new Amazon has improved very noticeably on its usability score for the Checkout phase – this is in part due to the folding of six previous stages onto three from 2001 to 2002, and also because of higher scores for layout of user options, condensed information presentation, and again more intuitive navigation to the next stage in the book purchase task.

3. High (R+C) criteria include user guidance, appropriate responses and the clarity of interaction messages and information presentation.

4. Low (R+C) criteria include the confirmation of the purchase and abort facilities.

5. Red-flagged (problem) areas based on the raw (un-weighted) scores from Table 17.4 are the lack of meaningful error messages, unexplained delays, and no intuitive undo facilities.

The results, and especially the approach adopted, namely that of prioritizing certain criteria over another set tailors the metrics to the nature of the task. On an intuitive level, it is clear that repetition of a task should make it more important (i.e. increase its weight); that the level of interaction required should also increase its weight; that the task duration should increase its weight; and that the level of knowledge required for the task should also increase its contribution to the website's overall usability score. A more formalized analysis for task repetition values could be obtained through a frequency count of the Knowledge Representation Grammars (KRGs) and Sequence Representation Grammars (SRGs) produced from a Task Analysis for Knowledge Description (TAKD) analysis [8]. The method used here is less complex since it simply counts the occurrence of these during a typical (shopping and browsing) interaction session.
Conclusion

E-commerce applications are usually strongly task-oriented, goal-directed, Web-based interactions, thus they lend themselves to the use of structured task analysis approaches. This chapter offers additional perspectives on Web-based tasks, and also introduces a way of using task-analysis to improve Web-based usability. A task-based weighting scheme for evaluating these sites is proposed. This extended evaluation metric scheme and provides a more finely tuned mechanism for assisting developers to improve usability of E-Commerce websites, since the user's task is included in the formulation of the guidelines. Our mechanism makes use of a novel usability metric prioritising scheme to yield information which can be used during both the design and maintenance phases. The prioritising scheme yields values which can be noticeably different from more commonly applied unadjusted values.

The approach as outlined here needs to be applied to a larger sample of E-Commerce sites. It is currently designed for websites which fit the LSD and Checkout model but the principle of scoring and then weighting usability metrics will make it suitable for other types of E-Commerce sites such as Internet-based banking. This will require fortifying or changing the metrics, by replacing and re-designing some of the items. Developing a faster questionnaire that can be delivered via the Web, and to which users rather than experts can respond, will also be beneficial, and may facilitate the partial automation of the questionnaires. It would also be necessary to obtain reliability information on the results obtained, by for example comparing the results with those obtained by a heuristic evaluation, or by comparing with the results from the other tools, such as web log analytics, mentioned previously. Websites are using increasingly sophisticated methods to watch what customers are doing and to make e-commerce pay – a growing future trend will be shoppers simulation – i.e. testing the usability against sophisticated computer models of the site [W9].

References


Web References


