Preprint: final version available from:

BLANDFORD, A., THIMBLEBY, H. & BRYAN-KINNS, N. (2003) Understanding Interaction Traps. In *Proc. HCI2003 Vol.2.* 57-60.

UNDERSTANDING INTERACTION TRAPS

Ann Blandford[1], Harold Thimbleby[1] & Nick Bryan-Kinns[2]

[1] UCL Interaction Centre
University College London
26 Bedford Way
London WC1H 0AP, U.K.
{A.Blandford,H.Thimbleby}@ucl.ac.uk

[2] Dept of Computer Science
Queen Mary University of London
Mile End Road
London E1 4NS, U.K.
nickbk@dcs.qmul.ac.uk

ABSTRACT

We propose the idea of an *interaction trap* as a consequence of dissonance (or "misunderstanding") between users and systems. These result in inefficient or failed interactions that, from an objective viewpoint, could have been efficient and successful. Interaction traps result in an unsatisfactory user experience and possible long-term inefficiency. We illustrate the argument with examples taken from both empirical studies (which demonstrate that such traps occur) and discussion of systems design (arguing that such traps can often be anticipated, and hence avoided, in design).

Keywords

Usability; Dissonant models; User experience, Interaction trap; Interaction barrier.

1. INTRODUCTION

Interactions between users and computer systems – whether traditional desktop machines or novel (e.g., ubiquitous) systems – have qualities that both the participants in the interaction and outside observers could comment on – for example, that the interaction was exciting, productive, frustrating or a waste of time. To be useful to designers, qualitative judgements have to be grounded in features of the design that can be understood and changed.

In this paper, we focus on a cause of a particular set of negative user experiences – namely frustration or failure caused by *interaction traps*. We initially identified traps in empirical studies: for example, as described more fully below, we observed users searching for information in digital libraries becoming convinced that the information they sought was not available and giving up, when in fact the information was there – they were just 'looking the wrong way'. An interaction trap is a situation where, due to the way features of the system design interact with the user's understanding of the system and their objectives, the user forms an incorrect understanding about the achievability of an objective, or about *how* to achieve an objective.

Many positive design principles and concepts have been proposed, including *modelessness* and *affordance*. Whilst nuances of these concepts are debated, the core ideas are not necessarily sophisticated, but attract attention because they are effective in design. Thus, "An affordance is an aspect of an object that makes it obvious how the object is to be used" [8]. Gibson [3] discusses "misinformation for affordances" – that is, features of the environment that lead users to misinterpret what is possible, such as doors with pullable handles that have to be pushed; such negative affordances are one source of traps.

Negative terms like *frustration* and *unsuccessful* reflect the fact that many interactions between users and systems are less than they could be. While much work now is addressed at creating a positive user experience – for example, in ideas of computers for fun [4] – there it still much work to be done on eliminating the negative. Introducing new terms (trap, barrier) allows us to introduce new distinctions. These concepts provide terminology for thinking about undesirable features of interactions which should, where possible, be designed out. However, unlike most existing approaches to assessing qualities of interactions, interaction traps are part of a framework that encourages the analyst to think about the broader context of the interaction – e.g. where users have multiple objectives, shifting objectives or interleaving tasks. This point is discussed more fully below.

2. TYPES OF INTERACTION TRAP

There are two main manifestations of interaction traps, which we can categorise as three types.

The first manifestation of a trap is that ***the user comes to believe that some objective is not achievable with the***

current system when in fact it is. There are two alternative outcomes:

- ***Type 1: The user has learned that the outcome is not achievable and ceases to try to achieve it***. This effectively cuts down the space of interaction possibilities because the user abandons the objective indefinitely, even into future sessions.
- ***Type 2: The user realises that an objective has become unachievable by the current strategy but would be achievable another way***. A user may be trapped for some time, but later discovers a way out. This is effectively 'type 1 with learning'.

The second manifestation is that *the* ***user believes that some objective is achievable from the current state when in fact it is not***. They waste time trying to achieve the unachievable. This situation generally leads to the outcome that…

- ***Type 3: The user correctly knows that an objective has become unachievable***. This is termed a barrier. A user may be trapped and *later* realise they have hit a barrier; an observer (perhaps with better insight into the design) has to be careful to distinguish 'type 1' traps and barriers.

Type 1 traps result in irretrievable breakdown in the interaction; types 2 and 3 result in *detours* – interactions that are much longer and less efficient than necessary. Here, we will not consider the uses of traps and barriers for security purposes – when a designer intends to put up barriers to some users.

Interaction traps arise through a breakdown in understanding between user and computer system. For traditional applications, of the kind we have studied, this will mean that the user misunderstands some key aspect of the system state or interaction possibilities. Sometimes, it can be viewed the other way – that the system has an incorrect representation of key user information (e.g., of their context or intentions). We use symmetric terminology – that there is a dissonance in models – to keep the argument symmetric between users and systems.

Although interaction problems have been long recognised, and recommendations made for avoiding them [9], there has been little design-oriented work on dissonance of models. There are general exhortations (e.g., [5]) to ensure high quality user feedback, and Norman [6] discusses the *gulf of evaluation* between user and system, but he does not unpack the idea. Reason [7] applies this notion in the context of human error in a discussion of latent errors – in which dissonance in models sets up a situation where an error is likely to manifest itself later in the same interaction. We call the point at which dissonance become inevitable the *point of deviation.* Interaction traps are not directly concerned with safety critical human error, but with interactional phenomena that manifest themselves as inefficient or unsuccessful interactions, as illustrated below. They are not a simple consequence of system design or of user knowledge, but emerge from the interplay between the two.

3. EXAMPLES

We became aware of interaction traps in a study of digital library use [2]. Here, we highlight selected incidents from those interactions to illustrate interaction traps. The study involved seven users, each of whom worked with a selection of commercially available digital libraries (such as the ACM Digital Library, Ingenta and EBSCO), searching for documents of their own choosing.

3.1 Failure to achieve an objective (type 1)

We start with examples of traps such that the user came to believe (incorrectly) their objective was unachievable. The first is taken from the interaction between user A and the New Zealand Digital Library. She had selected an article to view, but then read from the screen: "expanding the text here will generate a large amount of data for your browser to display". Taking the warning as implicit advice on something to avoid, she then typed "1" and selected "go to page." The result was that only page 1 was displayed. Having read this page, she commented that "I am looking for the option how to download, not just explore it." After 6.1 minutes in total, she had tried every option she could find for downloading the entire article, and concluded that "I've tried all, but I can't download."

The point of deviation occurred when A elected to view just page 1: at that point, the computer system limited the scope of what was possible in a way that she did not recognise. Most of the subsequent six minutes of interaction was devoted to achieving an objective (downloading the article) that was not, from that situation, achievable. This is an interaction trap: the user can only download and print pages that have been displayed. The trap had two distinct negative outcomes: the user spent six futile minutes trying to achieve something that was not achievable from that point, and the user left the interaction believing the objective to be in principle unachievable.

The second example comes from user E's interaction with the ACM digital library. Very early on in the interaction, E was looking for articles on digital libraries within the digital library (the library contains both journals and conference proceedings on the subject). The user's familiarity with search formulation in general led her to manually truncate (or stem) her search terms from "digital libraries" to "digital librar", commenting as she did so that that "If I put libraries I'm going to miss those with library so I'll put…" This resulted in no search results being returned. The ACM digital library allows the user to specify that the search engine should stem terms, but it assumes that the user has entered complete words for matching. The user did not find any articles on digital libraries in the ACM library through the entire interaction; she had apparently concluded that they either do not exist, or that they could not be accessed by such a search. In this case, the point of deviation occurred where the user chose to manually stem the term; the system has an incorrect representation of the user's objective, and its response of "no matches" did not alert the user to her misunderstanding, so the interaction proceeded with the user believing her objective to be unachievable.

Later in the same interaction, the user changed her objective, causing interference between system goals that she was unaware of. She was searching for articles related to artificial intelligence and HCI and entered "artificial intelligence human" into the subject search box. However, she had previously been searching for authors with surname Hollnagel, which was still in the author search box. Therefore the user interface was set up to search for articles containing the words "artificial intelligence human" and written by Hollnagel. The system's representation of the user's objective was again incorrect. Unsurprisingly, no articles were found. Attempts to reformulate the search did not improve matters: "What happens if I turn off the human and try again?… I would expect to get a fair amount. … No matches. Ha ha ha." The information about the author search term was available on the screen, and in a short interaction the user would have remembered that the term had been entered. However, in this case, we have an extended interaction, with a transition to a different objective and the user's attention now focused on the subject area of the screen. The information about the search formulation is not being adequately communicated by the computer system to the user. Again, this is an example of an interaction trap – the user now incorrectly believed the library did not contain articles indexed by the term "artificial intelligence."

These three interactions share features in common: each has a point of deviation, where a key aspect of the user's knowledge and objectives becomes different from that represented in the system state; each resulted in extended, futile interaction, and an outcome of each was that the user believed their objective to be unachievable. We can reflect on the causes of these deviations:

1. the user's choice – guided by the system warning message – to only show one page dramatically cut down the space of interaction possibilities;
2. a user misunderstanding about how stemming was implemented resulted in the system misinterpreting the user's objective;
3. a transition from one user objective to a different one was not recognized by the system.

While not all of these causes of deviations could be anticipated in design, it should be possible to anticipate some, and guard against them. For others, it is essential to give a quality of feedback such that the dissonance in models becomes quickly apparent and hence correctable.

3.2 Causing detours (type 3 traps)

The traps described above are all cases where the user, through the interaction, came to incorrectly believe that their objective was not achievable, and therefore stopped trying to achieve it at all. The converse case – that the user believes an objective to be achievable when in fact it is not – also occurred in the study. However, in this case, the user was forced eventually to give up and accept the non-achievability of the objective. In this case, the user experiences a detour, engaging in extended interaction that has no (or little) useful outcome. We consider this to be an interaction trap if the system displays some information that leads the user to believe that something is achievable, not when the user starts the interaction not knowing whether or not a library will contain particular material and subsequently finding that it does not, without ever developing a firmer belief that the material is available.

Our primary example is taken from the interaction between user B and Science Direct; other users in the study experienced very similar interactions. He was working with the Ingenta library at the time, and selected the "full text" option for an article. A message appeared to inform him that full text was available via Science Direct, which he therefore selected. When asked to enter a user name and password, he supplied those he had been given for Ingenta, failing to distinguish between the two: when the experimenter interrupted him after over 2 minutes of entering and re-entering the Ingenta user name and password, to tell him that he could not download the file, he asked "Why?" The significance of the transition from one document source to the other in terms of access rights was not clear to him. In this case, there was an important transition event, in which the user ceased to communicate with one library agent, and started communicating with a different one, without that difference being made salient to him. This is a point of deviation between user and computer system; again, the subsequent interaction is inadequately signposted for the user to immediately grasp the significance of the transition.

3.3 Beyond Digital Libraries (type 2 traps)

We have used digital libraries as our source of illustrations for traps of types 1 and 3. Traps of type 2 (where the user comes to believe that the objective is unachievable but then discovers a different strategy to make it achievable) were not found in our user studies: on no occasions did any of our users working with digital libraries spend a long time trying to achieve something one way then discover a different way of achieving it. This may reflect the fact that digital libraries are very complex systems, generally supporting very large spaces of interaction possibilities, but it also says something stronger: that the digital libraries included in our study did not support users learning – that once the user was trapped, there was no way out other than admitting defeat. To illustrate type 2 traps, we turn to more restricted and specialised systems.

One author of this paper encountered an interaction trap checking information on a web site. Wanting to know about future events, (s)he changed the default search period from being "today" to the end of the year to being today until the end of April. The system responded with "This date is not valid 04/31/03". Thinking that the problem must be to do with the different ordering of months and days on the two sides of the Atlantic, (s)he re-selected pull-down menu items to construct the date as 31/04/03 – same error message. Of course, the problem was that April only has 30 days. There was a difference between the objective as thought of by the user ("lectures until the end of April") and that understood by the system ("lectures until 31st April"). This problem is avoidable; it is also predictable, in that the system offers an option of a date that does not exist, but then bars it *once* the user has chosen it. The user had to draw on world knowledge about the calendar to work out a way out of the trap.

The Kegworth air accident, in which an aeroplane flew into the bank of the M6 motorway while trying to land at East Midlands Airport [1] is a safety critical example. There have been many extensive analyses of this incident; in summary, the left engine of the aeroplane failed, but the crew shut down the right (functioning) engine. They struggled with the problem for many minutes until the realised their mistake, by which time it was too late to take corrective action. There is a type 2 interaction trap: the crew believed that their objective was achievable by following a particular course of action, but the actions they took closed down the space of future interaction possibilities, and the feedback they received did not alert them to their misunderstandings until too late.

4. DESIGN CONSIDERATIONS

In discussing these examples, we have benefited from hindsight. It is much easier to recognise difficulties once they have occurred than it is to anticipate and prevent them. However, in our account, we have highlighted the design features that have led to the interaction traps and also made error recovery more difficult than it might be.

One obvious issue is the quality of feedback from systems such that users can understand why an action has had the effect it has, and hence recover more quickly to restore congruence of models. This is hardly a new insight, but as our examples show, a wide variety of systems score poorly on this criterion. In many cases, the point of deviation is some time before the point at which it becomes obvious to the user that the interaction is not proceeding successfully; better feedback on both the system state and the system's interpretation of the user's objectives are needed.

Many interaction traps are partly caused by the space of interaction possibilities changing abruptly as a consequence of a user action. Such traps arise from the opposite of the positive qualities of direct manipulation (incremental, reversible, etc). Thus, an objective that could be readily achieved from one state becomes more difficult to achieve – often requiring backtracking – from a subsequent state. These changes are poorly marked within the interaction, so that the user is typically unaware of the marked transition in possibilities. Some established systems such as word processors deal with the question of what options are available generally or from a particular point in the interaction by greying out and deactivating non-available options; other systems, such as web browsers, can lead users into traps because displayed options are uninformative or misleading. We have identified many cases where a system makes a possibility available to a user at one point, only for the user to discover some time later that that possibility does not exist. This arises because the system implementation does not take care to check that resources required later in an interaction are in fact available.

As discussed above, many systems appear to be designed for "single shot" interactions that have a single objective. Indeed, many evaluation techniques (e.g., Cognitive Walkthrough [10] and most laboratory-based approaches to empirical evaluation) focus on individual tasks. In practice, many interactions are supporting multiple objectives – whether in parallel (e.g., a user wants to achieve a higher score in a game than his previous best, but also wants to enjoy the interaction) or serially (e.g., the user looks for information on one topic then information on a different one). The consequences of users having multiple objectives, shifting objectives or interleaving interactions with multiple windows while multitasking are rarely considered fully in design, and yet these transitions can be a potent source of traps.

Interaction traps are a way to think about certain breakdowns in interactions between users and systems. These are interactions such that user knowledge and system model deviate to create dissonance, resulting in failure to achieve objectives and detours in the interaction. Traps are a property of the interaction, not solely a function of system design or user behaviour. The design challenge is to develop error recovery and repair mechanisms that empower users, reducing the frustrations and disappointments of detours and unnecessarily ineffective interactions.

5. ACKNOWLEDGEMENTS

This work was funded by EPSRC Grant GR/M81748. Harold Thimbleby is a Royal Society-Wolfson Research Merit Award Holder. We are grateful to Phil Barnard, Paul Cairns, Michael Harrison and Hanna Stelmaszewska for useful discussions about these ideas.

6. REFERENCES

[1] AAIB (1990) Air Accident Investigation Branch, *Report on the accident to Boeing 737-400.G-0 Brie near Kegworth Leicestershire on January 8th 1989, Department of Transport Aircraft accident Report 4/90,* London, HMSO.

[2] Blandford, A., Stelmaszewska, H. & Bryan-Kinns, N. (2001) Use of multiple digital libraries: a case study. In *Proc. JCDL 2001*, pp179–188. ACM Press.

[3] Gibson, J. J. (1986) *The Ecological Approach to Visual Perception*, Lawrence Erlbaum Associates.

[4] Monk, A. (2002) Fun, Communication and Dependability: Extending the Concept of Usability. In X. Faulkner, J. Finlay & F. Detienne (Eds) *People and Computers XVI.* Springer. 3–14.

[5] Nielsen, J. (1994) Heuristic Evaluation. In Nielsen, J. & Mack, R. eds., *Usability Inspection Methods*, New York: John Wiley*,* pp25–62.

[6] Norman, D. A. (1986) Cognitive Engineering. In Norman, D. A. & Draper, S. W. eds. *User Centered System Design,* Hillsdale NJ: Lawrence Erlbaum, pp31–62.

[7] Reason, J. (1990) *Human Error.* Cambridge: Cambridge University Press.

[8] Rosson, M. B. & Carroll, J. M. (2002) *Usability Engineering*, Morgan Kaufmann.

[9] Stewart, T. F. M. (1981) The Specialist User, in Shackel, B. ed., *Man-Computer Interaction: Human Factors Aspects of Computers & People, NATO Advanced Studies Institutes Series*, **E**(44), pp457–483.

[10] Wharton, C., Rieman, J., Lewis, C. & Polson, P. (1994) The cognitive walkthrough method: A practitioner's guide. In Nielsen, J. & Mack, R. eds., *Usability Inspection Methods*, pp105–140, New York: John Wiley.