THEA:
A Technique for Human Error Assessment Early in Design

Steven Pocock, Michael Harrison, Peter Wright & Paul Johnson

University of York, Heslington, York YO10 5DD U.K.

stevep|mdh|pcw|pdj@cs.york.ac.uk

Abstract: THEA is a technique designed for use by interactive system designers and engineers to help anticipate interaction failures. These may become problematic once designs become operational. The technique employs a cognitive error analysis based on an underlying model of human information processing. It is a highly structured approach, intended for use early in the development lifecycle as design concepts and requirements concerned with safety and usability – as well as functionality – are emerging. We believe the technique advances the systematic identification of human-computer interaction error through its straightforward application, requiring minimal formal knowledge of human factors or cognitive psychology.

Keywords: THEA, error analysis, cognitive failure, usage scenario

1 Introduction

Usability testing today plays an increasingly important role in system development as technology becomes more sophisticated. One popular approach, empirical usability testing, while comprehensive, is costly. It also takes place late in the design process, and requires at least a working prototype to be carried out. In response to such concerns, usability inspection methods have appeared. These methods take account of cognitive aspects of usability, based on underlying models of human cognition. Amongst these is the formative evaluation method known as cognitive walkthrough (CW) (Wharton et al, 1994) which formalises a way of imagining how operators think, and how they perform actions when encountering an interface for the first time.

THEA, the method described in this paper, possesses certain similarities with CW.

- Potential interface problems can be discovered early in design.
- An information-processing model underpins the method.

- Requirements can be identified and refined and applied successively to versions of the specification.
- Task descriptions and other aspects of context describe the situation in which the interface will be fielded.

A structured method is proposed for applying the method.

Unlike CW, however, a key driver for the development of THEA was for the technique to be carried out by system engineers who are likely to possess limited grounding in human factors. Difficulties reported with the practical application of CW, as described in (Spencer, 2000; John & Packer, 1995), highlight further differences between the two methods (see Table 1).

THEA has its roots in human reliability assessment (HRA) methods (Kirwan, 1994), but unlike these methods, it is specifically designed to inform human-computer interface design at an early stage of development. Situated between CW and HRA, it offers a finer granularity of analysis than CW and is more suggestive and easier to apply than HRA methods. The fact that an underlying cognitive model is embodied within the THEA error analysis questionnaire (described later), means that the need

for prior human factors experience or familiarity with cognitive theory is greatly reduced.

It is a strongly suggestive technique, guiding the analyst in a structured way to consider areas of a design for potential interaction difficulties. Other methods, such as the *human error identification in systems tool* (HEIST) described in (Kirwan, 1994), possess similar goals to THEA, except that THEA achieves them with considerably less exertion – 18 error analysis questions as opposed to 113 in HEIST, which is perhaps why this method has remained largely theoretical. The method, like (Galliers et al, 1999), is scenario driven. Their method however is focussed on constructing Bayesian Belief Networks (BBNs) to generate probabilistic data on error frequency
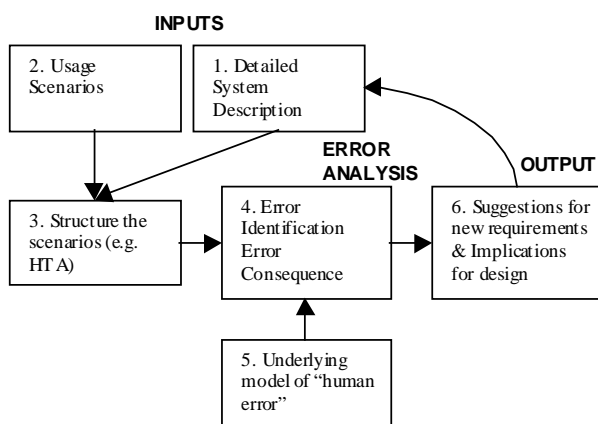

Figure 1: The THEA Process

2 THEA

THEA views errors as contextualised phenomena influenced by, for example, environmental factors. The technique, illustrated in Figure 1, considers contextual issues explicitly through the use of scenarios. In this way it is hoped to elicit how work is *actually* practiced rather than the way designers *envisage* it being carried out.

2.1   Method

THEA employs a systematic method of asking questions and exploring interactive system designs based on how a device functions in a scenario. This provides a structured means of critiquing a design and developing further requirements (Fields et al, 1997). In this way, it is hoped to help system designers anticipate human interaction failures which

may become problematic once a design becomes operational.

The technique is intended primarily for use early in the development lifecycle while functionality is emerging, and begins with a formal description of the work under analysis. This is achieved by combining a detailed description of the design under consideration – preferably with the assistance of domain experts – and a number of usage scenarios. This is illustrated in Figure 1, boxes 1 & 2 respectively. We now describe the process in more detail.

CW	THEA
Application focussed at the individual user level	Equal applicability at the individual user level and system integration level
Concentrates on ease of learning highly prompted interfaces through user exploration	Iterative analysis of how operator behaviour contributes to overall system dependability
Emphasis on problems involving information presentation and feedback	Equal importance of user goals, plans and actions as well as information presentation and feedback (perception, interpretation and evaluation)
Cognitive questions have sometimes been mis-interpreted	Tightly controlled, using an unambiguous questionnaire approach
Specifically considers whether user will select each of the correct actions along the solution path	The scenarios in which operator actions take place contain no solution path
No guidance for rating the frequency and severity of usability problems	THEA tool support requires analyst to enter a severity rating for each potential problem. A summary graph is output for each scenario.
Traceability can be problematic, especially if carried out over an extended period of time	Traceability a priority. THEA tool automatically handles data organisation and tracks all changes
Certain social constraints (e.g. lengthy design discussions and defensiveness) can hamper effectiveness	THEA specifically designed for use by designers and system engineers, ameliorating these constraints

Table 1: Some principal differences between cognitive walkthrough (CW) and THEA

2.2   Scenarios

THEA takes the view that "cognition is in the world" (Norman, 1988), that is to say, the context or

circumstances in which the actions are performed, is an important determinant of human performance. Through use of detailed scenarios, THEA analyses strive to capture the complex conditions which often result in humans interfacing with technology in unanticipated and unintended ways. Scenarios should thus comprise not only *actions* which take place in a given situation, but also *contextual factors* which surround the action, allow it to happen, and provide opportunities for "error". To represent the context as comprehensively as possible, a scenario template has been constructed (Fields et al, 1997) and is shown in Table 2.

Scenarios can be gathered by a number of means.

- Earlier versions of the system may provide experience of interesting situations. 'Top-down' designs are relatively infrequent and previous versions usually have associated reports highlighting problem areas.
- Incident and accident reports describe problematic situations.
- Accounts of frequent conditions and normal operation can be valuable.
- Situations where technology changes may be an important driver as in the case study presented here.
- Descriptions of situations that exemplify where concepts change may be important, for example, changing from voice-based communications in air traffic control to digital data-linking.

The question often arises as to how many scenarios are required to capture the usage context in sufficient detail. The answer relies on expert judgement as to when a 'good enough' coverage has been achieved, and for this reason it is highly desirable to have at least one domain expert involved in the scenario construction process.

2.3   Goal decomposition

To structure and interpret information contained in scenarios, Hierarchical Task Analysis (HTA) is a practical – but by no means the only – way of achieving goal decomposition (Figure 1, box 3). It is hierarchical because task goals are broken down into a structure of sub-goals which must first be achieved before the top level goal can be satisfied. In this way we can describe operators' tasks in terms of the goals and sub-goals to be achieved and the actions used to achieve them. Plans are appended to each task to describe the flow of control through the task, detailing how the sub-goals and actions within a task are combined to satisfy the higher level goal. HTA examples may be found in (Fields et al, 1997; Kirwan, 1994).

| Agents	- The human agents involved and their organisation - The roles played by the humans, plus their goals and responsibilities
Rationale	Why is the scenario interesting?
Situation & Environment	- The physical situation in which the scenario takes place - External & environmental triggers, problems/events which occur - What assumptions have been made?
Task context	- What tasks are performed? - Do formal procedures exist?
System context	What devices and technology are involved?
Action	- How are tasks performed in context? - How do activities overlap? - Which goals do actions correspond to?
Exceptional circumstances	How might the scenario evolve differently?

Table 2: A template for describing scenarios

Task descriptions, while good at describing what a user has to do and know, are less effective at describing how an interface might respond to a user's inputs. THEA presumes that some notion of causality can be used to explore the interaction between, for example, a display and other perceptual cues, operator memory requirements, and other aspects of the design. Guidewords based on a cognitive model of operator-system interaction (Norman, 1988) are employed. These can trigger questions about the extent to which, for example, a display is able to support goals and plans, or to consider how apparent it would be for an operator to perform an appropriate action. We believe this affords a means of linking task and system descriptions more directly, and forms the basis of the THEA error analysis phase.

2.4   Error analysis (EA)

The foregoing steps identify a number of factors facilitating an understanding of the context in which human actions – and therefore erroneous actions – take place. We are now in a position to draw these strands together in the error analysis phase (Figure 1, boxes 4 & 5).

As mentioned earlier, the analysis phase adopts a structured questionnaire-/checklist- style approach, referred to in (Fields et al, 1997) as the "Cognitive Error Analysis". This is based on failures (see Table 3) that are possible in Norman's execution-evaluation cycle model of human information processing (Norman, 1988).

THEA questions (see Figure 2) comprise four categories concerned with Goals (G1-G4), Plans (P1-P3), Performing actions (A1-A4), and Perception/Evaluation/Interpretation (I1-I7). A full list of the THEA error analysis questions is given in (Fields et al, 1997). These questions are actually specified as statements for which there is a true or false answer: *(I2) The effects of any changes on the system arising from the user system are perceivable immediately.* The questions have been derived by considering each stage of Norman's model and identifying questions that arise from each of the standard error 'phenotypes' (Hollnagel, 1998), see Table 3. In Table 3 the phenotypes (omission, commission, etc) are given words that are suggestive for each stage.
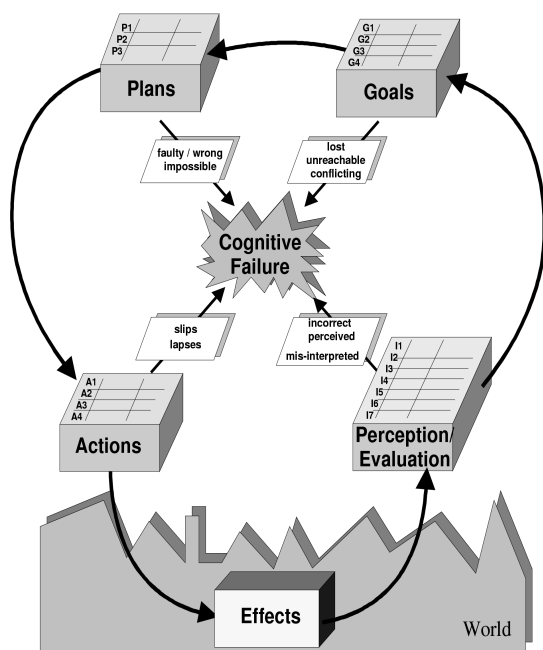
Figure 2: Cognitive failure in the context of Norman's 'loop' model and THEA.

Hence the question I2 was produced by considering a possible "omission" in the context of "perceiving" a change as a result of a user's action on the system.

2.5 Analysis questionnaire

The error analysis poses questions about the scenario to reveal areas of design where cognitive failures may occur, and assesses their possible impact on the task or system being controlled. A simple example might be the high level goal of photocopying a sheet of paper. One of the THEA analysis statements is: (G4) *if the task's main goal has associated sub-goals, these must first be accomplished before the main goal itself can be achieved.* For most photocopiers the analyst would typically answer "false" since it is entirely possible to walk away with your copy but leave the original document (and perhaps your copier card) in the machine. The sub-goal has thus been lost and a post-completion error has occurred. Conversely, a bank cash dispenser will not release money until the bank card has been withdrawn by the user. Such an interlock prevents the post-completion error of forgetting to remove the card.

Stage	Cognitive failure
Goals	Lost/Unachievable/Conflicting No triggering/activation Triggering/activation at wrong time, or wrong goal activated
Plans	Faulty/Wrong/Impossible
Actions	Slip/Lapse
Perception/ Interpretation	Failure to perceive correctly Misinterpretation

Table 3: Types of cognitive failure

When performing a THEA analysis, there will be occasions when no obvious behavioural manifestations are evident. For example, if an operator is presented with conflicting goals (G3), this may *itself* be the 'manifestation' of a problem which, if serious enough, may necessitate a design solution. Exactly how the analysis is carried out is largely a matter of choice, but two methods are envisaged.

1. The goal hierarchical structure is followed from top to bottom asking each question about *each* goal or action.
2. Parts of the scenario are selected where potential problems are anticipated. A detailed analysis of behavioural error and impact is conducted where appropriate.

The first option will clearly afford the most thorough analysis and is recommended for new designs. For

complex designs, the analysis will naturally be a lengthier process but is likely to uncover a greater number and range of concerns. With ProtoTHEA tool support, discussed briefly below, such complex analyses are facilitated.

3 Tool support with ProtoTHEA

As a result of conducting large and complex case studies, we identified a need for tool support to assist with the error analysis input and data handling. This resulted in the development of ProtoTHEA, a prototype tool where in addition to a representation of the error analysis questionnaire, scenario and HTA information for each project is entered via a graphical user interface. The tool's purpose is to support the method by providing easy input of scenarios and HTAs as well as matching the questionnaire to items in the scenario. For example, A1 checks that *"There is no mental or physical difficulty in carrying out this task"*. The analyst can answer 'True', 'False' (adding whether it is considered to be Low, Medium, or High severity), 'TBD' (to be decided, if no decision has been reached on this question), or 'N/A' if the question is not applicable for the current task. Space beneath each question allows for analysts' comments to be inserted concerned with "Causal Issues", "Consequences" and "Design Issues".

Hence, keeping a trace of the process as well as information about the completeness of the analysis is possible. All data is stored in a user-transparent database, and for each scenario an output in the form of a 'failure state profile' (adapted from (Reason, 1997)) is automatically generated. Such output is intended to flag error occurrences against phases of the Norman cycle (Table 3) thereby highlighting areas of the design which have been identified by the error analysis as potentially problematic. The tool also tracks analysis changes made during design review and update sessions.

4 Recording the results

Analysis results may be recorded according to project requirements. However, we have found that a tabular format provides a practical way of presenting the information. Table 4 shows a typical arrangement, while Table 6 provides an example for the first case study discussed next. This format is also consistent with other styles of error analysis prevalent in safety assessment and provides a useful object for external assessment of the results of the evaluation.

Question	Causal Issues	Consequences	Design Issues
Question identifier as an aid to traceability	Issues raised by analyst	Consequences of the causal issue	Notes, suggestions, comments, re-design ideas

Table 4: Tabular format for recording EA results

5 Application of THEA – case Study

We now illustrate a practical application of THEA by means of a case study, based on information collected from flight crew, involving a change of technology on the flight deck. A major change between the old and the new flight decks concerns the crew complement being reduced from three people to two, the flight engineer being replaced by computerised technology. The scenario involves a situation where the activities of the flight engineer would, on the old flight deck, be particularly significant. We deal with emergency conditions rather than normal operation, but since the tasks in themselves are fairly straightforward and do not involve much decision making, the crew activities involve more knowledge intensive activities such as fault diagnosis.

5.1   Situation and environment

The starting condition involves a four-engine fisheries patrol aircraft at low level over water, photographing a fishing vessel. To conserve fuel, the aircraft is flying on engines 2,3,4 only. Engine 1 (leftmost) has been closed down for fuel economy reasons. The aircraft suffers a massive bird strike on the right side. As a result of bird ingestion to engines 3 and 4, both engines fail producing engine failure and engine fire warnings. The engine problems will cause the failure of the generators in these engines, which will in turn lead to the remaining generators being overloaded, resulting in a series of warning or cautions being signalled after a short delay. This scenario may seem unlikely but is in fact taken from interviews with pilots who see this as a potentially serious problem on such missions.

5.2 Actions in context

As we discussed earlier, one of the principal components of a scenario is a description of the actions which take place. An HTA may certainly be employed, but it is not always necessary. If, for example, interaction with the system of interest is relatively simple, then it is probably sufficient to identify the goals that users have, and write down a list of the actions necessary to achieve the goals. If the interaction is more complex, then a more formal approach for capturing tasks and goals, such as HTA, may be required. As an alternative to HTA for this relatively simple scenario, we present one possible alternative representation, although it is not our intention to produce a fully worked example, rather to suggest how an alternative technique may be employed.

In Table 5 we show some of the crew and 'system' actions in the early stage of the scenario, with time flowing downwards. We can see the actions performed by each agent (both pilots and the 'system'), and we can observe both pilots conducting possibly contradictory actions at the same time. For example, the pilot is attempting to restart engine 1 to produce more thrust, while the co-pilot is shutting down the faulty engines i.e. reducing thrust. However, what such a diagram does not show are the links between actions and the surrounding context which is a main reason for thinking about scenarios in the first place.
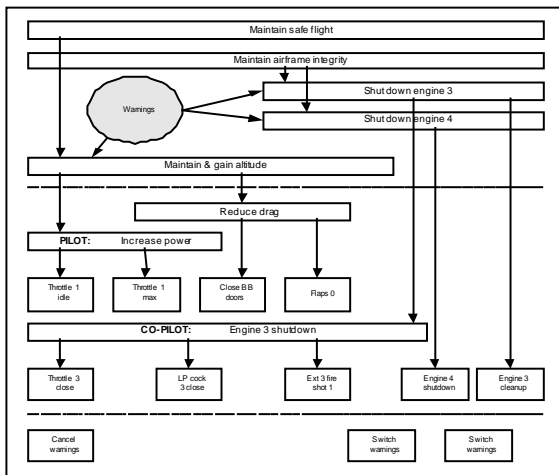

Figure 3: Goal structured scenario actions

As a useful alternative to Table 5, we could describe the actions of the scenario differently, such that we are able to see the order of task occurrence, and the goals (derived from the task analysis) to which they are directed. Figure 3 illustrates these relationships. Here, the same actions as Table 5 are shown, but additionally the goals that drive the interaction, as well as the triggers that bring the goals into being, can now be identified.

5.3 Error analysis excerpt

As mentioned earlier, error analyses may be conducted in two ways, either by systematically asking each question about each goal or action (applicable in high consequence systems), or through holistic application of the questionnaire to each scenario. We take the latter approach with the case study just described. Although the system we are analysing is a high consequence system, a full systematic analysis would be excessive here and an analysis extract is provided for illustrative purposes instead. Table 6 selects two of the questions from the full cognitive error analysis which are particularly pertinent to this scenario, namely:

- *A trigger exists for activating the task (via interface instrumentation or the environment)* (G1).
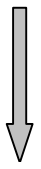- *There are no discernible goal conflicts associated with the task* (G3).

G1 yields a number of possible answers since different collections of goals have different triggering properties. Some are fairly innocuous and do not suggest potential problems (e.g. "Shut down engine" is triggered directly by a warning), whereas others are less directly triggered and may be more prone to being omitted (e.g. "Engine 3 cleanup"). The complete THEA analysis is presented in (Fields et al, 1997).

When conducting the full analysis, causal issues, that are raised and produce noteworthy or problematic consequences, are documented in the "Consequences" column. Entries for certain questions may be left blank. This indicates that the question does not appear to reveal any interesting insights. A third column entitled "Design Suggestions" might also be inserted. Thus we might append to G3: "Provide a display scheduling the goals that must be carried out, with a facility for dismissing or clearing achieved goals". In table 6 we omit the design suggestions column.

A statement of this kind provides a trace of a judgement that has been made. It is therefore possible that these judgements may be considered by other experts with more experience of likely problems and that together they may provide appropriate suggestions for redesign. THEA is a

qualitative technique. However, the rationale is based on a plausible model of how humans behave when interacting with the world. In the end, however, any judgements made are essentially subjective in nature. The point is to make these judgements as open to scrutiny as possible.

System status	Pilot	Co-Pilot	Info sources	System response
Engine 3 fire warning Engine 4 failure warning *Time*	Throttle 2 max. Press master warning Throttle 1 idle Throttle 1 max. Navigate safe exit route	Close ext. doors Flaps 0 Rudder trim Warn crew Throttle 3 close Engine 3 LP cock shut Engine 3 fire ext: shot 1	Airmanship Airmanship Engine 3 fire drill	Select 'ENG ECAM' page Start engine

Table 5: Scenario timeline showing actions performed by each agent

Question	Causal Issues	Consequences
G1 (Triggers, task initiation)	Many goals triggered fairly directly (e.g. "Shut down engine 3") Timing of lower level goals arises as a combination of triggering and group decision making (e.g. Engine 3 shutdown) Some goals rely on airmanship skills for their activation (e.g. power, drag) Some goals are poorly triggered, especially if there are several goals with only a single trigger on the display (e.g. "Engine 4 shutdown" or "Engine 3 cleanup").	Behavioural consequence is that triggers for cleanup actions exist in the display, but are removed when other tasks intervene. Switching to "Engine 4 shutdown" removes indications for "Engine 3 cleanup". It is also possible that "Engine 4 shutdown" or "Engine 3 cleanup" might be omitted or delayed.
G3 (Goal conflicts)	Goals to increase power and Engine 3 shutdown are in conflict (although this is inevitable)	Resolving conflict satisfactorily requires negotiation between pilot and co-pilot. The time required for this may lead to a non-optimal (too late) decision.

Table 6: Extract from the completed THEA error analysis

6 Discussion

Operators working within technologically sophisticated safety-critical domains such as nuclear power production, aviation or medicine, interface with systems possessing intricate defences-in-depth to reduce the likelihood of accidents. Yet accidents and incidents still occur despite such safeguards. To identify ways in which interfaces may be vulnerable to erroneous operator action, we maintain that some form of *qualitative* error analysis, as distinct from more 'traditional' quantitative HRA approaches, is essential. Much work has been carried out in recent years to model human-computer interaction breakdown, and methods, including the cognitive walkthrough, have been developed for this purpose. This approach attempts to identify specific areas in the interaction process where users are likely to encounter problems but, as we have discussed in this paper, some users have experienced difficulties in applying the method.

Additionally, the focus on ease of learning through exploration of highly prompted interfaces gives more information about user needs than specific interface usability problems. The subject of this paper, THEA, is a formative error analysis technique that builds on preceding approaches as a practical means for assessing system vulnerability to erroneous operator interaction. An important

antecedent of the THEA error analysis process is gaining an understanding of how the system being examined will, or may, be used in practice. Thus we formulate 'usage scenarios' to provide us with context of use – the circumstances or conditions under which an event occurs – to elicit how work will *actually* be performed as opposed to how it is *envisaged* it will be performed. It is highly desirable to carry out the analysis early in the design lifecycle before adverse consequences may be encountered at the 'sharp end', and before a design becomes 'rigid' and excessively difficult or expensive to modify. We differentiate between cause and consequence since incorrect operator actions and assessments are treated as the *starting* point for analysis rather than the conclusion – they are recognised as *symptoms* rather than causes. In this predictive role, causes are the initiating events and manifestations are the possible outcomes. Of course, THEA works equally well for retrospective analyses of extant designs.

We have found from experience that although no special expertise is required to carry out the error analysis procedure, input to the process from domain experts significantly expedites its completion. Additionally, tool support offered by ProtoTHEA has demonstrated an ability to manage large and complex case studies. Whether the 'traditional' or tool-assisted approach is employed, the emphasis of THEA is on functionality and practicality. We believe it offers a significant step forward in error analysis, being straightforward to use, and does not require specific training in human factors or cognitive psychology. Indeed, the technique has been specifically designed for use by system engineers and design teams.

A recent case study involving a design for a new rocket launch platform, employed the technique to appraise the new platform operator interface where specific erroneous actions could result in damage to the platform as well as serious injury to the operator and other crew personnel. THEA identified areas of the design which might contribute to erroneous operator interaction, and also provided an assessment of possible consequences.

Our results corroborated the clients' own numerical analysis thus affording a more confident design assessment. A further benefit encountered was the convergence of system engineers and human factors personnel through the exchange of ideas and techniques, helping to overcome what (Hollnagel, 1998) describes as "*the conceptual impuissance or abstruseness*". THEA has recently been successfully used (Cartmale & Forbes, 1999) by the UK National Air Traffic Services (NATS) to analyse procedures

for a major software upgrade to the Air Traffic Control (ATC) system at the new Swanwick Centre. The technique is currently in use with BAE SYSTEMS and has also been applied in-house on case studies involving aircraft fuel systems and engineering maintenance tasks.

References

Cartmale, K. & Forbes, S. (1999) Human error analysis of a safety related air traffic control engineering procedure. In *People in Control: International conference on human interfaces in control rooms, cockpits and command centres.* IEE, University of Bath, UK.

Fields, B., Harrison, M. & Wright, P. (1997) *THEA: Human Error Analysis for Requirements Definition.* Technical Report YCS 294. Department of Computer Science, University of York, York YO10 5DD, UK.

Galliers, J., Sutcliffe, A. & Minocha, S. (1999) An impact analysis method for safety-critical user interface design. *ACM Transactions on Computer-Human Interaction.* 6(4) pp. 341-369.

Hollnagel, E. (1998) *Cognitive Reliability and Error Analysis Method CREAM*, Elsevier Science.

John, B. E. & Packer, H. (1995) Learning and using the cognitive walkthrough method: a case study approach, In *Proceedings of CHI'95 (Denver CO*, May 1995), ACM Press, 429-436.

Kirwan, B. (1994) *A guide to practical human reliability assessment*, Taylor and Francis.

Norman, D. A. (1988*) The Psychology of Everyday Things*, Basic Books.

Reason, J. (1997) *Managing the Risks of Organizational Accidents*, Ashgate Publishing.

Spencer, R. (2000) The streamlined cognitive walkthrough method, working around social constraints encountered in a software development company. *CHI Letters* 2: 353-359. ACM Press.

Wharton, C., Rieman, J., Lewis, C. & Polson, P. (1994) The cognitive walkthrough methods: a practitioner's guide In *Usability Inspection Methods* (Eds, Nielsen, J. & Mack, R.L.) Wiley, New York, pp. 105-140.