

Unmasking Mode Errors: A New Application Of Task Knowledge Principles To The Knowledge Gaps In Cockpit Design.

Hourizi, R & Johnson, P

Human Computer Interaction Laboratory,
Computing Science Group
University of Bath
Bath BA2 7AY
UK

R.Hourizi@bath.ac.uk & P.Johnson@bath.ac.uk

Abstract: In the complexity of modern, computerised systems, the current system state can come as an unpleasant shock to even an experienced user. The dominant thinking in this field [Palmer, 1995, Palmer et al, 1993, Degani et al, 1996, Rushby 1999] suggests that much of this "automation surprise" [Palmer 1995] comes as a result of mode error. In this paper, we will show that this is a fundamental misclassification and show the existence of a deeper problem, which we identify as a task knowledge gap [Johnson 1992] between operator and system. We suggest a new design solution, based on this re-classification and evaluate this new approach through the redesign of interfaces in a simulation of a well-documented air-incident. The results of our evaluation show the redesign to be significantly less prone to the depicted error. We conclude by considering the implications of these results in terms of task knowledge principles.

Keywords: Aviation, Safety, Automation Surprise, Mode Error, User Interface Redesign, Task Knowledge Structure

1 Automation Surprise

In the complexity of modern, computerised systems, the current state of the system can come as an unpleasant shock to even an experienced user. This problem, known as "automation surprise" [Palmer, 1995] can have extremely serious consequences

A primary area where for well-documented accounts of serious and fatal consequences arising from such situations is the field of aviation. That will, therefore be the focus of this study.

2 Air Inter Flight F-GGED

One such situation is the incident, involving Air Inter flight F-GGED between Rheims and Strasburg

Though the causes of such a crash are inevitably complex, there is agreement [Bureau Enquetes Accidents, 1992, De Keyser & Javaux,

in eastern France:

At 18:20 hours, on the 20th January 1992, an Airbus A320 passenger jet, flying a short journey from Lyon to Strasbourg (Air Inter flight F-GGED), crashed into the foothills of Mont St-Odile in eastern France, killing all but six of the passengers and the entire crew. When the surviving cockpit voice recorders were retrieved from the wreckage, it was found that the crew had shown no sense of panic during the final moments of the flight, had attempted no evasive action and had, apparently, been entirely unaware that a crash was imminent, until an altitude alarm had sounded 200 feet above ground level – too late to avoid impact. No mechanical failure could be identified as leading directly to the crash, nor was any significant human malpractice to blame. [Bureau Enquetes Accidents, 1992]

1996] that inappropriate entry to the autopilot was central to the events leading to the disaster.

In order to understand this erroneous entry, a basic understanding of the plane's Flight Control Unit (FCU) is needed:



Figure 1: Airbus A320 FCU (Original) [Meriweather, 2000]

Essentially the FCU is made up of switches, which affect the entry format of related bi-modal dials. An example is the Speed/Mach dial, which allows plane speed to be entered as either a number of multiples of the speed of sound (Mach) or as a number of feet per minute (Speed), depending on the setting of the Speed/Mach switch. Parameters can be entered either individually or in combination (e.g. go to an altitude of 15,000 feet at a vertical speed of 400 feet per second).

An autopilot setting is then chosen, using one of six buttons towards the bottom of the FCU. An example would be the AP 1 button, which engages Autopilot 1, after an automatic delay of four seconds.

The theoretical task procedural structure for the "enter instruction to autopilot" task (derived from the accident report [Bureau Enquetes Accidents, 1992] and Meriweather's [2000] instructions for use) can, therefore be summarised as per Figure 2.

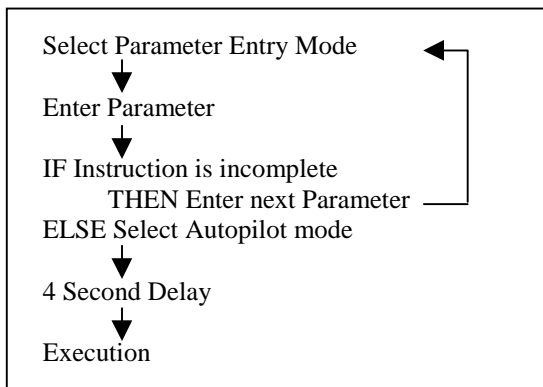


Figure 2: Instruction Entry Task (Original Interface)

In the case of flight F-GGED, the pilot and co-pilot had been refused permission to land on their first request and, were about to make a second attempt using navigation assistance from the local air traffic control. At the time of the error, both pilot

and co-pilot were extremely busy, needing to correct their lateral course, get the landing gear down, run through the pre-descent checklists and enter an appropriate rate of descent.

The overloaded pilot, entered a speed of 3300 feet per minute in the place of an angle of 3.3 degrees (Both are entered through the bimodal VS/FPA dial). The desired entry would have given a relatively gentle descent, whilst the error lead to a steeper drop, culminating in the plane crashing into a mountainside short of the airfield. A schematic description of this incident is included in Figure 3, below

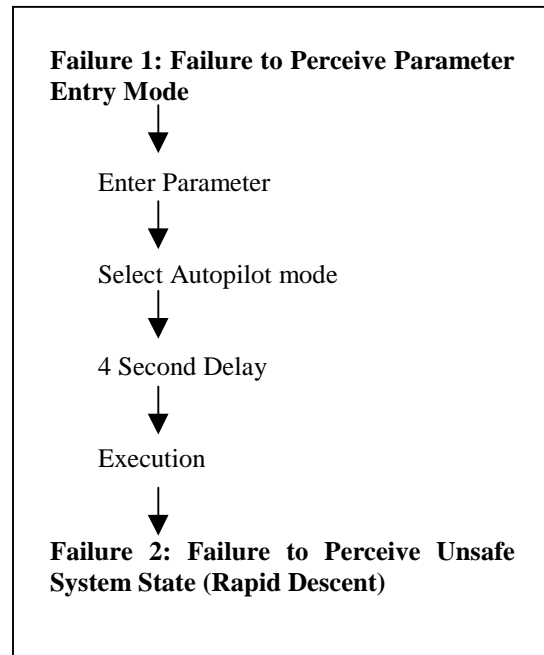


Figure 3: Erroneous Instruction Entry, Air Inter Flight F-GGED

3 Mode Error

Since many of the examples of cockpit automation surprise involve similar failures to Failure 1 (above) i.e. they involve a failure to perceive the current system mode, the dominant descriptive model [Palmer, 1995, Palmer et al, 1993, Degani et al, 1996, Rushby 1999] of the surprise lists mode error, or the mis-identification of current mode, as a key causal factor in the accident.

If we examine these modal errors, as described, they fall under the category of perceptual slips i.e. the pilot (or operator) fails to notice the current mode of the system and, therefore, ends up with a system which reacts to his/her input in an unexpected way.

Having identified a slip as the pivotal issue in the cause of an accident, many authors (sensibly) go on to describe solutions, which involve the prevention of that slip. For example, Rushby [1999] builds models, which identify systems and interfaces that are “prone” to such errors, suggesting individual features, which need reworking. Palmer [1995] suggests lock-ins, to prevent the passage from mode-confusion to hazardous state.

4 The Problem with Mode Error

The mode error approach does not, however explain the second failure described in figure 3, the misperception of the unsafe system state after the erroneous entry. In the case of flight F-GGED, the pilots had information which could have led them to understand that a sustained, rapid drop was taking place both from the instrument panel, had they chosen to consult it and from the raw physical sensation of quick descent. They took neither into account. De Keyser [1996] describes this as “fixation error”, an error during which the pilots are presented with considerable evidence that the system is not performing the intended action but continue to believe that they have correctly grasped the system mode

Given that perceptual slips are not generally followed by confirmation bias in other domains [Reason, 1990], their regular occurrence in cases of automation surprise suggests that a more fundamental element exists which links the two “failures” described above. This element would, if extracted, provide a better explanation of the surprise than mode error alone.

5 Knowledge Gaps

In fact, this category of failure is a much better match with those described for knowledge-based errors than for perceptual slips [Reason 1990]. Operators are hampered by both confirmation bias (a tendency to confirm an existing world view in the face of contradictory evidence) e.g. the 1988 Air France crash at Mulhouse Habsheim [Degani et al 1996], where the pilots continued to fight the joystick, even though it was not affecting the flight path as expected – and selectivity (focus on only those factors which support the current world view) e.g. the pilots on flight F-GGED, described above, ignored the physical sensation of a rapid loss of altitude in favor of their mental model which told them that they were on a slow, controlled descent.

In this context, mode errors become a symptom of the underlying problem (the operators lack of knowledge of the current system state), rather than the cause - a relationship which explains why they are inevitably surrounded by other, complementary errors (selectivity and confirmation bias).

6 A New Solution

Having redefined the problem at the core of automation surprise, we must now find a corresponding solution. If these are indeed knowledge errors, we might expect the task knowledge principles for human computer interaction [Johnson, P, Johnson H and Hamilton, F, 2000] to offer guidance towards a potential solution.

As per Huettner [1998], the pilots role has changed from a simple hands on operator, to that of process manager, co-coordinating the efforts of a series of interconnected agents, some of whom, like the co-pilot, will be human and, therefore exhibit considerable flexibility and situation awareness and others, like the autopilot, which will be more limited in scope but are still able to affect the system state. We call this the coordination task.

It should be noted at this point that we have no intention of implying intelligence on the part of the autopilot, nor, by extension that interaction between pilot and computer should occur on the same level as that between pilot and co-pilot. However the understanding of a part of the system, which can affect the position and momentum of the plane is key.

We can now look for examples of successful approaches to the coordination tasks within the cockpit and use them as the basis for our interaction between human and computer agents.

7 Modeling Cockpit Interaction

In the highly formalized environment of flight control, there are many examples of such coordination, including that between ATC and the cockpit, pilot and co-pilot and even cabin crew and flying staff. Unlike the relatively newer problem of the human-computer interface, these interactions have existed for many decades and have the benefit of research, implementation, re-examination and correction over that time.

The benefit to the designer is the fact that a coordinated approach has been developed to the transfer and checking of information, formalizing

the passage of instructions between human agents involved in the task of flight control. The aim of this approach (detailed in the following sections) is both to update the knowledge base of the actors in each role (flying pilot, non-flying pilot and ATC) and to check that this knowledge base has been synchronized after each instruction. This is achieved through the use of acceptance, repetition and confirmation procedures.

If our class model is correct, we should be able to apply this formal approach to the interaction between pilot and autopilot and thereby significantly affect the occurrence of automation surprise.

The next step is to develop and test this assertion - a step we achieved by returning flight F-GGED.

8 Sources

In order to redesign the interface, we need a task model for the transfer of coordinated instructions within the cockpit. As per Johnson [1992], a starting point in this analysis is the identification of appropriate sources for the study of goals, tasks, subtasks and procedures. Unfortunately, a number of approaches (interviews, questionnaires, direct observation) were not available, since we had no direct access to pilots or ground crew.

On the other hand, the accident report of flight F-GGED does include a transcript of the conversations between flying pilot (PF), non-flying pilot (NFP) and Air Traffic Control (ATC) during the final 30 minutes of the journey [Bureau Enquetes Accidents, 1992]. Though a somewhat narrow basis for study, we can be re-assured that the commission of enquiry (including experienced aviators) made only minor criticisms of the interaction amongst the crew/ATC. We can therefore conclude that we have a relatively reliable source of indirect observation of a flight deck, which can be considered for our analysis.

In order to avoid the omission of key details, which could, by chance, not to show up on these (fairly short) transcripts, we have also cross-checked our results with other sources (the accident report and the Palmer, Hutchins, Ritter, van Cleemput study on altitude change procedures [1993]). These other sources also provide valuable input to our model.

Analysis of the interaction between ATC and cockpit and between pilot and co-pilot yielded the task structure described in Figure 4.

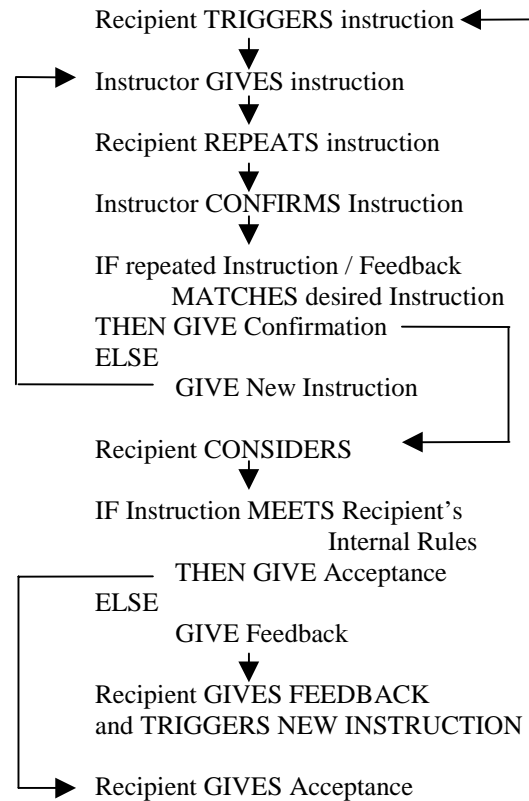


Figure 4: Task Model: Coordinated Cockpit Instruction Transfer

9 Analysis

A comparison of the task models described in figures 2 (Original FCU) and 4 (Cockpit Instruction Transfer) shows that the task of using the original interface is significantly different from the equivalent intra-cockpit task:

- No full repetition of the desired instruction is provided
- No confirmation is required from the pilot – the instruction is automatically executed after a four second delay
- No internal rule check is performed by the recipient (autopilot). Simple errors, such as “fly to a dangerously low altitude” (e.g. into the ground!) are therefore not subject to any lockout.
- The feedback given to the instructor (pilot) is both distributed i.e. given parameter by parameter and passive i.e. the instructor has to make an effort to look for it.

These omitted elements constitute the core of the coordination task outlined in section 7, above. The

principle of procedural dependency (the reliance on familiar task procedural structures during interface usage) [Johnson et al 2000] would suggest that their inclusion in a redesigned interface would reduce the underlying knowledge gap and, therefore, the system error rate.

10 Redesign

Our next task was, therefore, to match the workflow required by the interface to the underlying task structure.

Since this task structure was essentially linear (i.e. it contained relatively few conditional branches), we opted for a linear design, where the user started at one side of the interface (we chose the left) and worked across to the other (the right). This gave the following blueprint:

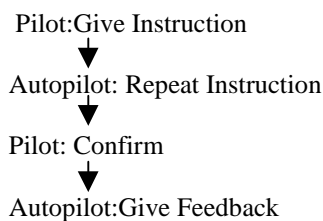


Figure 5: Redesigned Task Model

We then modified this model to give the autopilot a single “voice” i.e. we collected all autopilot output into a single area. The strict linear interpretation of figure 5 would have called for at least two areas of output, the first for repetition and the second for feedback. However, prototype designs showed this to be confusing for users and we simplified the design.

This left us with the schema described in figure 6:

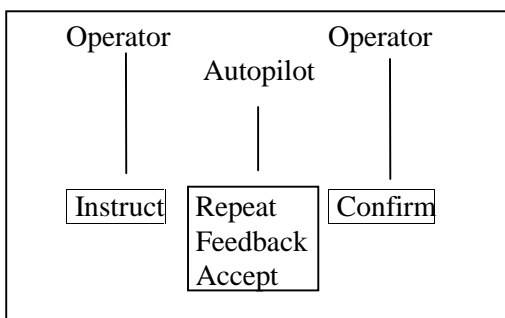


Figure 6: Abstract User Interface Design.

The final step was to flesh out the design. Many details were refined according to the usability

principles of user control, consistency, forgiveness, perceived stability, feedback, modelessness and progressive disclosure. The most noteworthy design decision, however, was to make the entry options surrounding each parameter more explicit by grouping mutually exclusive choices in columns. The users task was now represented as a journey from left to right, with a selection to be made from each column passed.

The final design is shown in figure 7 below.

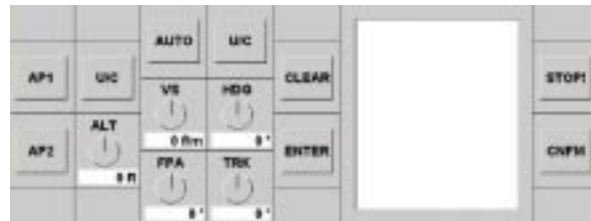


Figure 7 Airbus A320 FCU (Redesign – final result)

11 Usability Testing

In order to analyse the improvements (if any) offered by our redesign, we ran usability tests on the two interfaces (original and redesigned). Conscious of the potential risk of unrepresentative results obtained by an artificially created scenario, we returned again to the transcripts of flight F-GGED.

Though we were unable to test the interfaces on commercial pilots within the time frame of the project, a group of ten regular but non-professional computer users were found who would give sufficiently reliable results for an initial evaluation.

First, we created a condensed version of the transcripts by eliminating long periods of silence and all casual conversation, which had no bearing on the navigation of the plane.

Next we recorded the abbreviated version onto tape in order to ensure that each subject would receive identical instructions with consistent timing across the tests.

We divided the ten subjects into two equal groups. The first were trained on the original interface, before attempting to enter the taped instructions in real time, whilst the second underwent the same process, but using the redesigned interface.

The subjects were observed during the simulation and their entries for each individual instruction were recorded. In this way, we could measure and analyse not only the raw success/failure of each instruction, but, in the case

of an error, the false entry sequence. Finally, we recorded the time taken for each instruction entry.

A narrow measure of success was to be the elimination of the entry sequence, which played such a prominent part in the crash of flight F-GGED, but we also considered a wider set of results, including the number of errors made on each instruction, each dial and within each of a set of pre-determined categories.

12 Data

Within the narrow measure of success (the elimination of the specific error which caused the crash on flight F-GGED), we had remarkable success. 40% of the test group, asked to use the original interface to enter instructions, made the same inappropriate error as the actual pilot, leading to a theoretical crash.

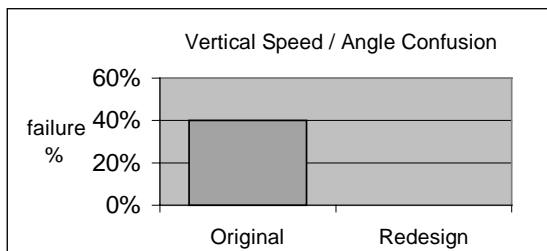


Figure 8: Vertical Speed / Flight Path Angle Confusion Errors by Interface

The second test group (those using the redesign), however returned zero occurrences of this mode confusion, supporting the idea that the redesign had achieved its most obvious purpose.

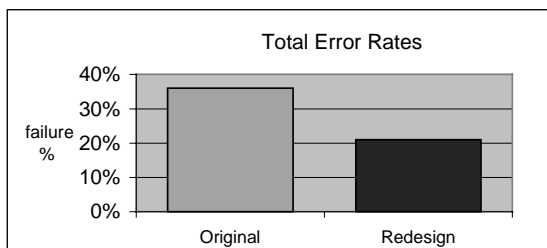


Figure 9: Summary of Total Error Rates by Interface

Behind this rather crude measure, however, a more detailed examination of the results also gave cause for optimism, though, rather inevitably, further work will be needed before a definitive conclusion can be reached.

Overall, error rates were reduced by over 40 percent and entry times also fell by 10 percent. This combination of more accurate entries, with no corresponding drop in input speed suggests that our interface redesign had helped to diminish the task knowledge gap identified above.

Equally encouraging is the spread of error rate reduction. Drops were seen across the different entry parameters (see figure 10 below), supporting the view that we had not simply solved a problem within the use of a particular dial or button but had rather improved the usability of the entire interface.

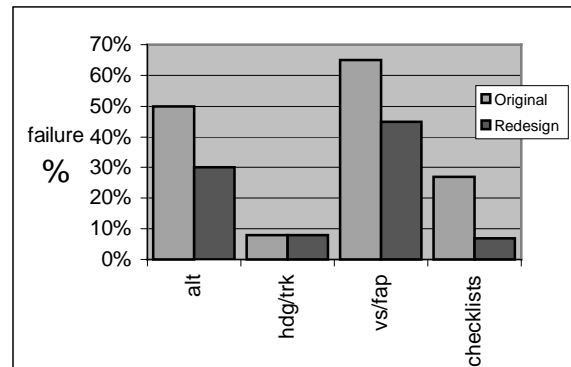


Figure 10: Error Rates by Entry Parameter

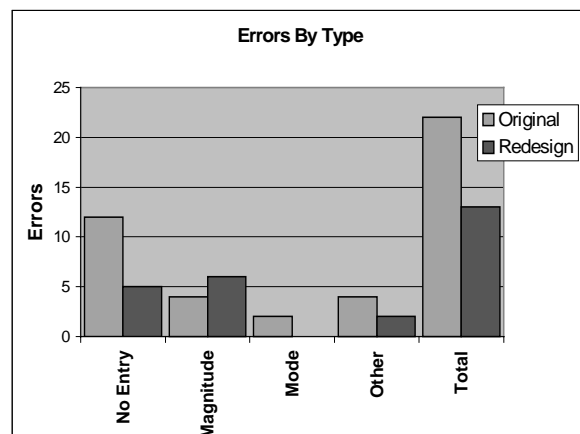


Figure 11: Number of Errors by Type

Having examined recorded errors by entry parameter, we next grouped the errors into the following categories:

- No Entry: The subject attempted no entry for the given instruction
- Magnitude Error: An error was made in the order of magnitude of a given parameter e.g. an altitude of 100 feet in the place of 10,000 feet
- FPA/VS Confusion: A vertical speed was entered in the place of an angle of descent

(flight path angle) – see flight F-GGED accident description above.

- Other: Errors not falling into any of the previous three categories.

In three of the four categories, we found marked improvement with our redesign (see Figure 11 above). A potential design solution for the fourth category will be discussed below.

13 Discussion

The residual error rate, even on our re-designed interface remained at just over 20 percent.

Though the types of error observed could, to some extent, be explained by the operator's unfamiliarity with the domain, we were unhappy with the idea of dismissing them, on this basis alone. For this reason, we developed three further elements for future inclusion in our interface environment, which, we believe, would remove the larger error groups. A short summary of these elements is as follows:

1. For the "No Entry" errors observed, subjects gave an inability to remember the complete instruction in a "rushed" environment as a major causal factor. Though the use of a tape to provide instructions has many advantages, it does have the drawback of limited interactivity, removing the state-check-confirm procedure seen between instructor and recipient in the real cockpit. For this reason, we intend to adjust our test environment to include repetition of instructions before parameter entry, believing that by focusing attention in this way we will aid subjects ability to remember their goals.

2. Next, whilst looking at the group of magnitude errors, we found that our subjects encountered great difficulty with the language of the cockpit. Notably, they confused target "levels" (given in hundreds of feet) with absolute altitudes (given in feet). Though it could be argued that this would not be a factor amongst commercial pilots, who would be familiar with such language, we felt that further amendments to the interface could make the issue clearer, even for experienced pilots, thus reducing the likelihood of such errors occurring. We will therefore change the input displays to emphasize the link between the two formats. Eg 12,000 feet (a level of 120) could be portrayed as "**120** 00".

3. We also feel that the set procedures surrounding the coordination task could be extended to encourage the pilot(s) to revisit feedback information before confirmation. Examples of such

an extension could be the reading aloud of an entered instruction for acceptance by a colleague or the autopilot.

14 Conclusions

In conclusion, we have examined the "Mode Error" approach to automation surprise and found it to be based upon a miss-classification of the human factors involved. We go on to assert that it is more insightful, in these situations, to say that co-ordination problems give rise to a knowledge gap on the part of the user.

By looking at other, relevant Air Traffic Management procedures, we have identified a generic co-ordination task procedure, designed to avoid the occurrence of such a knowledge gap.

Applying the task knowledge principle of procedural dependency [Johnson et al 2000], we then produced a new design solution, which by facilitating and enforcing a process of checks and confirmations supported this co-ordination task. In this way, we addressed the cause of the surprise (knowledge deficiency), rather than its symptoms (amongst others, mode error).

We then tested the new interface empirically, using a simulation of a well-documented air incident. We found that the new interface supported more accurate and faster instruction entry and would, therefore, have lessened the likelihood of the accident, had it been both available and in use.

This enhancement to cockpit safety demonstrates both the dangers inherent in interfaces designed on the basis of inappropriate task structures and the corresponding importance of appropriately designed interfaces .

Acknowledgement

Professor Johnson's research on this project is supported by EPSRC grant PRIDE: Task Related Principles for Interface Design and Evaluation.

References

- Bureau Enquetes Accidents, 1992,
Rapport De La Commission D'enquête Sur L'accident Survenu Le 20 Janvier 1992 Près Du Mont Sainte-Odile (Bas Rhin) À L'airbus A 320 Immatriculé F-Gged Exploité Par La Compagnie Air Inter.

- De Kayser, V, Javaux, D, 1996,
Human Factors In Aeronautics, *Design, Specification
& Verification Of Interactive Systems 1996*.
- Degani, A, Shafto, M, Kirlik, 1996,
Modes In Automated Cockpits: Problems, Data
Analysis And A Modeling Framework, *Proceedings
Of The 36th Israel Annual Conference On Aerospace
Sciences*
- Fields, R, Harrison, M, Bramwell, C, 1996,
Exploring Design Options Rationally, *Sigchi Bulletin
April 1996*
- Hutchins, E, Holder B, Hayward, M, 1999, Pilot Attitudes
Towards Automation,
[Http://Hci.Ucsd.Edu/Hutchins/Attitudes](http://Hci.Ucsd.Edu/Hutchins/Attitudes)
- Johnson, P, 1992,
*Human Computer Interaction – Psychology, Task
Analysis And Software Engineering*, McGraw-Hill
- Johnson, P, Johnson, H, Hamilton, F 2000,
In Schraagen, Jm, Chirman, Sf And Shaun, VI (Eds)
Getting The Knowledge Into Hci - Theoretical +
Practical Aspects Of Task Knowledge Structures
- Johnson, C, Love, L, 2000,
Using Diagrams To Support The Analysis Of System
'Failure' And Operator'Error',
www.Dcs.Gla.Ac.Uk/~Johnson/Papers
- Johnson, C, 1999,
Reasons For The Failure Of Crm Training,
www.Dcs.Gla.Ac.Uk/~Johnson/Papers/Crm
- Meriweather, J, 2000,
Airbus A320 Flight Deck,
www.Meriweather.Com/A320
- Huettner, C 1998,
The History Of Aviation Safety,
[Http://www.Aero-Space.Nasa.Gov/Library/Ch1.Htm](http://www.Aero-Space.Nasa.Gov/Library/Ch1.Htm)
- Palmer, E, 1995,
"Oops It Didn't Arm", A Case Study Of Two
Automation Surprises, *Proceedings Of The Eighth
International Symposium On Aviation Psychology*
- Palmer, E, Hutchins, E, Ritter, R, Vancleemput, I, 1993,
Altitude Deviations: Breakdowns Of An Error
Tolerant System, *Nasa Technical Memorandum, July
1993*
- Reason, J 1990,
Human Error, Cambridge University Press
- Rushby, J, 1999,
Using Model Checking To Help Discover Mode
Confusions And Other Automation Surprises, *3rd
Workshop On Human Error, Safety, And System
Development, Liege, Belgium, June 1999*