02WAC-98

DESIGNING USER-INTERFACES FOR THE COCKPIT:

FIVE COMMON DESIGN ERRORS AND HOW TO AVOID THEM

Lance Sherry
Honeywell

Peter Polson
University of Colorado

Michael Feary
NASA Ames Research Center

ABSTRACT

The efficiency and robustness of pilot-automation interaction is a function of the *volume of memorized action sequences* required to use the automation to perform mission tasks. This paper describes a model of pilot cognition for the evaluation of the *cognitive usability* of cockpit automation. Five common cockpit automation design errors are discussed with examples.

INTRODUCTION

The introduction of automation on the flightdeck of modern airliners has contributed to improved range, performance, and safety (Funk, 1997). Whereas this automation has reduced the *physical workload* of the pilot, it has increased the *cognitive workload* of the pilot (Woods, Johannesen, Cook, & Sarter, 1994; Billings, 1997; Federal Aviation Administration, 1996; Bureau of Air Safety Investigation - Australia, 1999; Air Transport Association, 1999).

Investigations by researchers of modern flight-deck operations have identified the complexity of learning and using cockpit automation. Pilots described the experience of learning to use this automation as “drinking from a fire hose” (BASI, 1999), and only achieve skilled and efficient use of the system after 12 to 18 months of line experience (Polson, Irving, Irving, 1994). Several studies and surveys of pilots have consistently revealed that pilots have difficulty in using the features of the automation during line operations due to “gaps in their knowledge of how that automation works” (ATA 1999, FAA 1996). These and other studies cited the need for more training (BASI, 1999; Feary, et. al. 1998, Hutchins, 1994).

In a study of the cognition required to perform 102 mission tasks using the B777 Flight Management System (FMS) and it’s Multi-Function Control and Display Unit (MCDU), Sherry, Polson, Fennell, & Feary (2002) found that 74% of the these tasks required training of *memorized action sequences* to complete the task. This directly contributes to the “fire hose effect” during training. Also, 46% of the tasks that were classified as occurring infrequently (i.e. less than once in every 20 flight legs) required the *recall of memorized action sequences*. Infrequent use of memorized action sequences results in erosion of pilot’s skill (Javaux, 2000), and directly contributes to perceived complexity during line operations.

This paper describes a model of pilot cognition that is designed to be used by engineers developing cockpit automation to maximize the efficiency and robustness of the pilot-automation interaction. At the root of this approach is the *minimization of memorized action sequences* that must be trained and then recalled during line operations. This paper discusses five classes of user-interface characteristics that lead to training and recall of memorized action sequences:

(1) Input devices require significant **reformulation of the mission task** into sub-tasks or alternative representations in order to use the automation

(2) Absence of labels, prompts, and/or organizational structure, require pilots to remember action sequences to **access** desired input devices (or information) in the hierarchy of cockpit displays

(3) Absence of prompts that define the **format** for data entry require pilots to memorize correct formats

(4) Absence of labels or prompts to identify how and where to **insert** the entry

(5) Representations and content of feedback displays require significant mental calculation or memorization to infer the intentions of the automation and to **verify and monitor** the long-term effects of the current commands.

The next section summarizes the model of pilot cognition used to design efficient and robust cockpit user-interfaces. The following sections summarize the five design characteristics with examples from the MCDU user-interface. Techniques are described to overcome these user-interface deficiencies.

PILOT PERFORMANCE AND COGNITION

A model of pilot's cognition for studying aviation pilot-automation interaction, developed by Sherry, Polson, Feary, & Palmer (2002), is illustrated in Figure 1, and summarized below. Pilot cognition is described by five discrete steps (**R**eformulate, **A**ccess, **F**ormat, **I**nsert, and **V**erify & Monitor). These steps are referred to as "RAFIV" in the remainder of the paper.

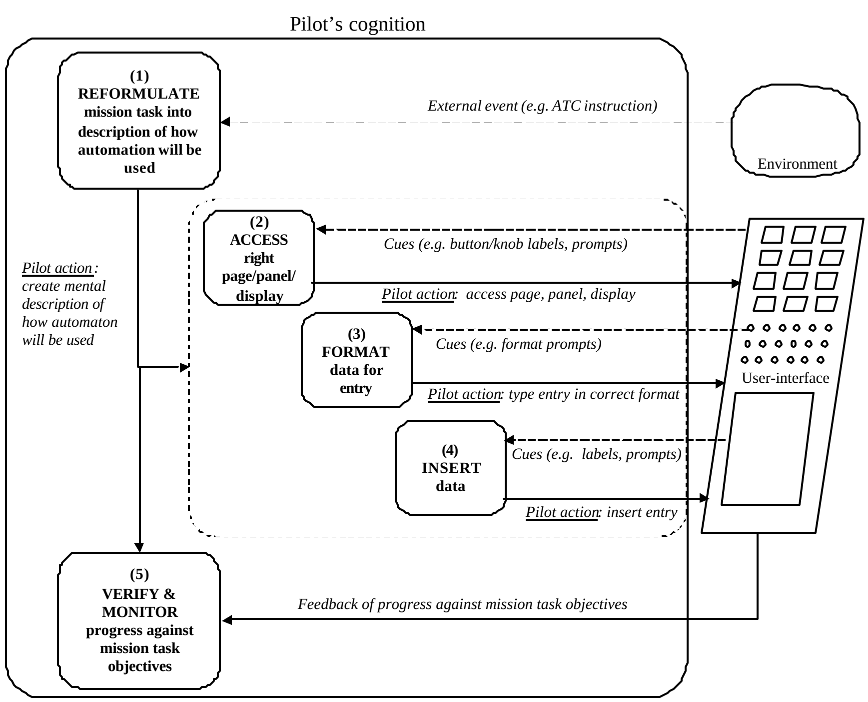
(1) ***Reformulate*** *the mission tasks into tasks and data that can communicated to the automation*. Pilots create a mental description of the how the automation will be used to perform a given task. For example an ATC clearance must be converted into a set of data that can be entered into the automation (Palmer, Hutchins, Ritter & van Cleemput; 1992).

Once a description on how to use the automation has been defined, the pilot must perform actions to transfer the description to the automation via a sequence of actions. These actions have been divided into three steps (Polson, Irving, Irving, 1994):

(2) ***Access*** *the right user-interface:* Once a description on how to use the automation has been defined, the pilot must access the right page (e.g. hierarchy of MCDU pages), panel (e.g. Mode Control Panel), or display (e.g. multi-function synoptic displays). The access step identifies the actions that must be taken on the user-interface to display the fields for data entry (e.g. Vertical Revision page on the Airbus) or orient pilots attention to the correct input device (e.g. Mode Control Panel LNAV button).

(3) ***Format*** *Data for Entry:* Once pilots have formulated the information to be entered into the displayed page, altitude widow, dialog box, etc., the pilots must format and enter the data (e.g. MCDU scratchpad typing). For example, the entry of a lateral route offset is <Side L or R><distance in nm.>. The format step described here is more specific than the Designate step of the Polson, Irving, and Irving (1994) model.

(4) ***Insert*** *Data:* Once the data is formatted the pilots takes actions to insert the data in the correct



Model of pilot cognition is represented by five steps: Reformulate task, Access display, Format data, Insert data, and Verify and Monitor. RAFIV model is used for cognitive usability analysis.
Figure 1

location. For example an entry in the MCDU scratchpad is inserted by selecting the line select key adjacent to the MCDU page field for the entry.

Once the entry has been made and the automation commands the aircraft trajectory, the pilot must verify and monitor the progress of the aircraft trajectory to satisfy the mission tasks goals input to Step 1. See Hutchins, 1994; Prevot, 2000, Jacobsen et. al. 2000.

(5) ***Verify & Monitor:*** The pilot must verify that the automation has: (1) accepted the pilot entry, (2) is performing the intended task within the envelope of acceptable performance, and (3) the task is satisfying the mission goals (Fennell, 2002). This step involves scan and intensive scrutiny of the PFD, ND, and MCDU.

PILOT PERFORMANCE

Each of the RAFIV steps required to complete a task is performed by the pilot by either *recalling* the appropriate action from long-term memory or by *recognizing* the appropriate action from salient visual cues in the environment such as, button labels or prompts on the user-interface (Polson, Sherry, in preparation).

✍ <u>Training Time for Recall Steps</u>: Steps of the RAFIV model that rely entirely on the recall of memorized action sequences for completion (i.e. steps without any visual cues) require *2 to 10 times* more time to train to competence than steps with visual cues (Kieras, 1997). Once the memorized action sequence is described, repetitive drill and practice is required to master the skill.

✍ <u>Reliability of Recall Steps</u>: Steps of the RAFIV model that rely entirely on the recall of memorized action sequences and are performed infrequently, will exhibit *less than 50% probability* of completion (Franzke, 1995; Kitajima, Soto, Polson, 1998). “If you don’t use it, you will lose it.”

FIVE COMMON DESIGN ERRORS AND HOW TO AVOID THEM

This section describes each of the five classes of design errors with examples, and strategies to avoid this phenomenon.

(1) INPUT DEVICES THAT REQUIRE REFORMULATION OF THE MISSION TASK INTO SUB-TASKS OR ALTERNATIVE REPRESENTATIONS

The most usable automation provides direct features for the completion of mission tasks. When the automation does not directly support the task, the pilot must reformulate the task into alternative tasks or a sequence of sub-tasks that the automation can perform (Palmer, Hutchins, Ritter & van Cleemput, 1992). This behavior relies on the use of memorized actions. This is time consuming and attention demanding, and therefore subject to increased training times and reduced reliability.

Tasks Supported by the MCDU/FMS	**Tasks <u>not</u> Supported by the MCDU/FMS**
? Alignment of ADIRU Position ? Flightplan/Route Planning ? Aircraft Performance Computations ? Direct To ? Holding Pattern at PPOS ? Lateral Route Offset ? Missed Approach/Go Around ? Descend Direct ? Descend Now	? Climb through intermediate altitude constraint ? Descend to crossing restriction ? Change departure/arrival runway ? Adjust climb speeds to achieve desired climb gradient ? Crossing radial with altitude restriction

Sample of tasks supported and <u>not</u> supported by the MCDU/FMS (From Sherry, Polson, Feary, & Palmer, 2002)
Table 1

Table 1, from Sherry et. al. (2002), list examples of mission tasks that are, and are not, supported directly by the FMS. For example, tasks that are supported directky include: Direct To a Waypoint (enter waypoint ICAO identifier into Line Select key 1L on the MCDU LEGS page), Hold at Present Positon (Hold Page), and Descend Now (Line Select Key 6R on the DEScent MCDU page).

In contrast, the basic mission task to descend to cross a waypoint at a specified altitude and speed cannot be performed directly by the automation. [Note: entry of a speed and altitude constraint at the specified waypoint in the flightplan does not guarantee that the aircraft will be commanded on an appropriate trajectory.] Instead the pilot must compute and command the required rate-of-descent using distance (or time) to the waypoint, ground speed, and altitude remaining. Furthermore, the pilot must determine the appropriate combination of airspeed, vertical speed, airbrake setting, pitch attitude and/or thrust to achieve the desired rate-of-descent and maintain within the safe operating envelope of the aircraft.

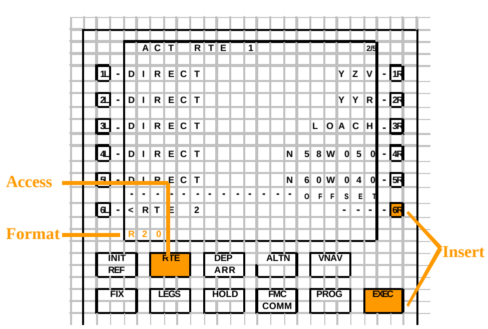
The only way to mitigate this class of design error is to *understand the mission tasks* and *provide automation to support the pilot in executing these tasks*. Several researchers (e.g. Vakil & Hansmann, 2000) have proposed including the mission task analysis as the starting point of the design process. For example, Riley (1998) designed a cockpit user-interface to accept Air Traffic Control commands as inputs.

In addition to the definition of the mission tasks, a critical element of the design process is the definition of the internal representation of the environment and the mission that is held by the automation. For example, the LEGS and RTE lists represent the flightplan as a sequence of legs and a sequence of procedures and airways respectively. The list structure of these representations determines the manipulations required by the pilot to make flightplan changes. Vicente (1999) provides a structured approach for deriving internal representations using a control engineering paradigm; (a) understand the plant (i.e. environmental constraints), (b) identify potential sources of disturbances, (c) define control objectives (i.e. mission goals), and (d) provide the means to control to these goals.

(2/3/4) ABSENCE OF VISUAL CUES FOR ACCESS, FORMAT, AND INSERT

Once the description of how to use the automation has been formulated, the pilot must access the correct input device (step 2), format the entry (step 3), and insert the entry (step 4). These manipulations have nothing to do with the standard aviate, navigate, or communicate tasks, yet constitute a large portion of the knowledge required by pilots to operate a glass cockpit. For example, 35% of the mission tasks using the FMS required memorization of action sequences to access the correct MCDU page (Sherry et. al., 2002). Memorization of format for data entries was required for 29% of the mission tasks.

For example, the three steps for the "lateral offset of the flightplan" task on the B777 are illustrated in Figure 2. The pilot reformulates the mission task by *recalling* that the offset can be accomplished using the automation by entry of the offset side (left or right) and offset distance to the FMS route. The reformulation step cues the pilot to access the RTE page using the RTE Mode Key on the MCDU. The pilot must then *recall* that the entry is formatted <side L or R> <distance>. This format is not prompted on the display, nor is it common with any other MCDU tasks. Finally, entry is inserted in the LS key appropriately labeled "OFFSET". In observation of this task on the line, it has been observed that pilots cannot remember the exact format of the entry and using trial-and-error converge on the correct format. This is time



MCDU RTE page for activation of the Lateral Route Offset task. Access, format, and insert actions are identified.
Figure 2

consuming and erodes pilot confidence in using the automation.

Access of the correct page (or simply location of the correct icon on a cluttered display), format of the entry, and insertion of entry are serious problems even for applications with well designed graphical user interfaces such as Microsoft Office. Selecting a menu item or clicking on a tool bar icon in a Windows environment requires some use of recall to remember that the function exists and the location of the icon. Likewise, format and insertion of data can rely entirely on memory. These steps are particularly difficult for novel or infrequently performed tasks.

Access

The cockpit provides several visual cues to aid these steps. Functions are accessed by the input devices on separate panels associated with a class of task. For example, the Mode Control Panel (MCP) is trained as the location for tactical flight path changes. The panels on modern cockpits do not provide strong visual breakdown of tasks for either aviate/navigate/communicate, or for the manual/tactical/strategic control of any aircraft axis.

Format and Insert

Knobs and wheels provide robust entry mechanisms. Typed entries on the MCDU are less robust. MCDU fields provide excellent visual indications for format and insertion (e.g. FROM/TO, COST INDEX, …etc) Other MCDU fields fail to provide useful format information. Most of these are associated with multiple entries with abbreviations such as the format for the lateral route offset entry (<side><distance>), and the format for step altitude constraint entries (/<altitude>S).

The best user-interface design for format and insert is the use of dialog boxes and pull-down menus that allow selection from a list of options without any typing. Both Airbus and Boeing MCDUs use "pull-down menus" successfully for stringing flighplan Runways, SIDS, STARs, and Approaches. Also the ability to select waypoints from the flightplan for insertion for Direct To is accurate, fast and eliminates errors introduced by typing. Abbott (1997) describes the application of dialog boxes and wizards for formatting and insertion of data for the MCDU.

(5) REFORMULATION OF DISPLAY FEEDBACK FOR MISSION TASK VERIFICATION AND MONITORING

The RAFIV loop is closed by the verification and monitoring of the intentions and future commands of the automation by the feedback displays of the automation. Like the mission task reformulation described above, verification and monitoring is best accomplished by direct verification and monitoring of representations of the mission task. For example, the green arc displayed on the Navigation Display (ND) during descent provides direct feedback of the future trajectory of the airplane and reflects the mission task of crossing a waypoint at an altitude. Conversely, the absence of feedback display of the relative position of current and future energy envelope of the airplane does not guide the pilot in making sound selections in the use of additional drag (i.e. airbrakes) or thrust setting to adjust the green arc to make the descent restriction. Instead the pilot must rely on conservative descents that avoid the energy envelope limits, and heuristics for the long-term trajectory effects of adjusting, pitch, thrust, and drag.

Like the reformulation of the mission task, described above, issues with verification and monitoring are derived from ambiguities in mission tasks and fidelity of the internal and visual representations of the environment provided by the automation. Vicente (1999) provides guidelines on how to create operationally meaningful visual representations that relate directly to the mission tasks.

CONCLUSION

The design of the functions supported by cockpit automation and their user-interfaces is optimized when the each step of the interaction with the automation requires the *minimum number of recalls of memorized action sequences* by the operator.

Efficient designs, resulting in the minimum required recall of memorized action sequences, for the reformulation of mission tasks (step 1), and the verification and monitoring of mission tasks (step 5), can be achieved by providing internal and visual representations of the environment that can be manipulated to achieve mission tasks.

Efficient designs, resulting in the minimum required recall of memorized action sequences, for the access (step 2), format (step 3), and insert (step4) have little to do with traditional aviate and navigate tasks, and are defined based on the structure and style of the user-interface. Characteristics commonly associated with "graphical user-interfaces" such as direct manipulation, menus, dialogue boxes, and pull-down lists, limit the amount of memorization required to complete the task.

NEW GRAPHICAL USER-INTERFACES FOR THE COCKPIT

Graphical user-interfaces alone, in the cockpit do not inherently improve the performance of Reformulation, Access, Format, Insertion, and Verify & Monitor steps. Instead it is the careful design of the functions in support of the mission tasks.

Several of the features generally associated with graphical user-interfaces invoke the recognize (not recall) paradigm. Graphical user-interfaces encourage visual representations of the environment (e.g. graphical flightplans). When these representations can be mapped directly into the environment (e.g. ATC instructions, aero charts) the reformulation required is minimized.

The other major characteristic of graphical user-interfaces is the application of pull-down menus, dialog boxes, and wizards. These mechanisms significantly simplify and eliminate errors in the access, format, and insert actions.

USABILITY ANALYSIS

It should be noted that the success of any new user interfaces for the cockpit lies in the abilities of the designers *to understand the mission tasks and provide automation to support the pilot in executing these tasks*. Once this has been accomplished, the design of the user-interface should address Access, Format and Insert issues. Several techniques have been developed to analyze the recall of memorized action sequences using office automation (Wharton, Rieman, Lewis, & Polson, 1994) and using cockpit automation (Polson & Smith, 1999; Sherry,.et. al., 2002).

ACKNOWLEDGMENTS

This work was supported by the NASA Aviation Operations Systems Program contract GS09T01BHM0386 order ID 9T1N001MH to Honeywell International CAGR (TPC: Michael Feary) and Cooperative Agreement NCC 2-904 with the University of Colorado. Special thanks for several technical suggestions to Everett Palmer, Mike Matessa (NASA-ARC), Randy Mumaw, Dan Boorman, (Boeing), Karl Fennell (United Airlines), Steve Quarry, Phil Scandura (Honeywell). We also thank Everett Palmer and Rose Ashford (NASA), John Kilroy and Jim Martin (Honeywell) for their support.

Honeywell Publication # C69-5370-020

REFERENCES

Abbott, T (1997) *A Comparison of Two Control Display Unit Concepts on Flight Management System Training*, NASA Technical Memorandum 4744.

Air Transport Association (1999) *Potential Knowledge, Policy, or Training Gaps Regarding FMS Generation Aircraft.* Automation Sub-committee, Human factors Committee, Air Transport Association. www.air-transport.org.

BASI (1999) Advanced Technlogy Safety Survey Report. *Flight Safety Digest Special Issue.* Flight Safety Foundation, June-Aug 1999, pages 137-216. Available at www.basi.gov.au/sprog/advtek.htm (9/00)

Billings, C. E. (1997) *Aviation automation: The search for a human-centered approach*. Mahwah, N.J.: Lawrence Erlbaum.

Feary, M., D. McCrobie, M. Alkin, L. Sherry, P.Polson, & E. Plamer (1998). *Aiding Vertical Guidance Understanding*. NASA/TM 1998-112217.

Federal Aviation Administration (1996) *The interface between flightcrews and the modern flight deck system.* FAA Human factors Team. Available at www.faa.gov/avr/afs/interface.pdf

Fennel, K. (2002) personal communication.

Franzke, M. (1995). Turning research into practice: Characteristics of display-based interaction. *Proceedings of human factors in computing systems CHI '95* (pp. 421–428). New York: ACM.

Funk, K. (1997) *Aircraft accident rates: conventional vs. advanced technology. Available at* www.flightdeck.ie.orst.edu/FDAI/hullLossrates.html *(11/29/00)*

Hutchins, E. (1994) An integrated mode management interface for training. Training automatin workshop, NASA-ARC, Moffet Field, CA.

Javaux, D. (2000) Assessing and understanding pilots knowledge of mode transitions on the A340-200/300. In *Proceedings of HCI-Aero 2000,* 27-29 September, Toulouse: France

Jacobsen A, S. Schen, J Wiedeman (2000) Vertical Situation Awareness Display. At Situation Awareness on the Fligtdeck: Avionics Mini-conference, *Royal Aeronautical Society*, London, March 23, 2000

Kieras, D. (1997) A Guide to GOMS Model Usability Evaluation using NGOMSL. In *Handbook of Human-Computer Interaction.* Eds, M. Helkander, T.K. Landauer, P. Prabhu. Elsevier Science B.V.

Kitajima, M., Soto, R., and Polson, P. G. (1998) LICAI+: A comprehension-based model of the recall of action sequences. In F. Ritter & R.M. Young (Eds.) *Proceedings of Second European Conference on Cognitive Modelling* . pp. 82-89. Nottingham University Press: Nottingham.

Palmer, E., E. Hutchins, R.D. Ritter, I. VanCleemput (1992) *Altitude Deviations: Breakdowns of an Error Tolerant System*. NASA/TM DOT/FAA/RD-92/7

Polson, P. & L. Sherry (in preparation) Approximate Models of Pilot Performance. University of Colorado.

Polson, P., & Smith, N. (1999) The cockpit cognitive walkthrough. *Proceedings of the Tenth Symposium on Aviation Psychology.* (pp. 427-432) Columbus OH: Ohio State University.]

Polson, P. S. Irving, J. Irving (1994) Applications of Formal methods of Human Computer Interaction to Training and Use of the Control And Display Unit. Tech Report 94-08, University of Colorado.

Prevot T. (1998) A display for managing the vertical path: An appropriate task with inappropriate task feedback. In *Proceedings HCI-Aero 2000 International*, Montreal, Canada.

Riley, V. (1998) Cockpit control language: A pilot centered avionics interface. In *Proceedings HCI-Aero International Conference on Human Computer-Interaction in Aeronautics*. Montreal, Canada.

Sherry, L. P. Polson, M. Feary, & E. Palmer (2002) *When Does the MCDU Interface Work Well? Lessons Learned for the Design of New Flightdeck User-Interfaces.* Honeywell Publication C69-5370-0021.

Sherry, L., P. Polson, K. Fennell, and M. Feary (2002) *Drinking from the Fire Hose: The over-reliance on memorized action sequences for MCDU operation*. Honeywell Publication # C69-5370-022.

Vakil, S. & J. Hansman (2000) Approaches to Mitigating Complexity-Driven Issues in Commercial Autoflight Systems. The 3rd Workshop on Human Error, Safety, and System Development *June 7-8, 1999* Liege, Belgium

Vicente, K.J. (1999) *Cognitive Work Analysis*. Mahwah, New Jersey: Lawrence Erlbaum Associates

Wharton, C. J. Rieman, C. Lewis, & P. Polson (1994) The cognitive walkthrough method: A practitioners guide. In J. Nielsen and R.L.Mack (eds) *Usability Inspection Methods.* New York: John Wiley.

Woods, D.D., L. Johannesen, R. Cook, N. Sarter (1994) Behind human error: cognitive systems, computers, and hindsight. *Wright Patterson Air Force Base*, Dayton, OH:CSERIAC.

CONTACT

Lance Sherry - lance.sherry@honeywell.com
Peter Polson - ppolson@psych.colorado.edu
Michael Feary - mfeary@mail.arc.nasa.gov