Automated Tool for Task Analysis of NextGen Automation

Lance Sherry, Maricel Medina, George Mason University, Fairfax, VA
Michael Feary, NASA Ames Research Center, Moffet Field, CA
Captain John Otiker, Southwest Airlines, Dallas, TX

Abstract

The realization of NextGen capabilities will require rapid deployment of revised airline cockpit procedures and the pre-requisite training and proficiency checks. Traditional approaches for the evaluation of the re-designed procedures and training, such as expert reviews and human-in-the-loop tests, cannot provide comprehensive analysis, cannot be performed until after the procedures and training are developed, and are cost and time prohibitive.

This paper describes the emergence of a new class of tools to automate the evaluation of procedures and training. The tools capture the procedures and tasks to be trained in a formal model that is stored in a data-base. Human performance models are executed to estimate the ease-of-learning, ease-of-use and likelihood of failure of each of the tasks. The procedures and tasks can be defined rapidly, and modified and run repeatedly throughout the development cycle. The underlying models and tools are described in this paper. A case study and the implications of these tools are also discussed.

Introduction

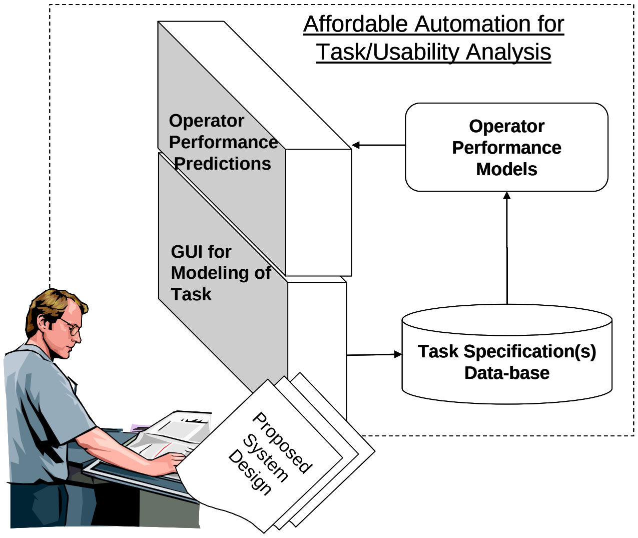
The realization of NextGen capabilities, such as data-link communication, self-separation/station-keeping, 4-D trajectory navigation, will require the rapid addition of new functions in the automation found on the flightdeck of commercial aircraft. The addition of these functions also requires the revision of airline Operator Procedure Manuals (OPM), as well as additional training curriculum and materials.

The revision of procedures and training is a significant expense that must be borne by the airlines that operate in a cost-sensitive marketplace. Furthermore, modification of procedures and training increases risk of accidents, incidents and reduces operational efficiency.

To ensure the safety and efficiency of the new procedures and training, airlines conduct intensive evaluation. Traditionally three approaches have been used. First, subject-matter-experts review the procedures and training materials and provide criticism based on experience and lessons learned. This subjective approach has been shown to be effective in certain settings (Dumas, 2003), but can be biased by group-thinking and does not ensure a comprehensive, systematic review (Turner & Pratkanis, 1998).

Second, airlines routinely conduct human-in-the-loop experiments to evaluate the procedures and the training along the lines of traditional Human Factors Usability tests (e.g. Nielsen, 1993). Whereas this is the most accurate approach, this approach is cost prohibitive and can only be conducted at the end of the development cycle when the cost of revisions is highest.

Third, more recently, airlines have conducted


Affordable Automation for Task/Usability Analysis
Operator Performance Predictions
GUI for Modeling of Task
Operator Performance Models
Task Specification(s) Data-base
Proposed System Design

Affordable automation for evaluation of procedures and training.

Figure 1

formal usability inspections, such as participatory design (Muller & Kuhn, 1993), cognitive walkthroughs (Wharton, Rieman, Lewis & Polson, 1994), and heuristic evaluations (Nielsen & Mack, 1994). This approach enables inexpensive evaluation early in the development cycle. Although this approach has been shown to be effective, they have not been widely used due to the absence of trained practitioners in the aviation industry. Further, these approaches also exhibit poor inter-rater reliability (Hetrzum & Jacobsen, 2003) due to differences in granularities of the task definition and the differences in the subjective ratings.

A new generation of tools for automated task analysis is reaching maturity that has the potential to overcome the issues with the approaches described above. The intent of these new tools is to provide domain experts (e.g. training developers, pilots) the ability to rapidly, repeatably, and comprehensively conduct evaluation of procedures, tasks, and training material. One such tool is CogTool (John, Prevas, Salvucci, and Koedinger, 2004). Another is HCIPA, the subject of this paper.

These tools provide the evaluator an easy-to-use graphical user-interface to capture the human-computer interaction required to perform a procedure or task. The procedure or task is stored in a data-base and can be revised for modifications to the procedure or training. The procedure or task in the data-base can then be analyzed using human performance models (e.g. KLN, ACT-R, CORE, empirical models) to estimate the performance of operators learning and performing procedures and tasks. See Figure 1.

These tools are fast, accurate, and provide the means to iteratively conduct a comprehensive analysis of all the tasks. They also eliminate the two sources of poor inter-rater reliability by capturing actual end-user button pushes (to eliminate ambiguity in the task definition), and by bounding the variance in human performance.

This paper describes a tool that is intended for use by aviation domain experts without formal Human Factors training to conduct HCI usability analyses. Specifically, the tool enables designers and testers to very rapidly assess the Trials-to-Mastery (i.e. time to competence for training) and the Probability of Failure-to-Complete, for each task called out in the procedures.

The paper is organized as follows: Section 2 provides an overview of the human-computer interaction process and the HCIPA model. Section 3 describes the tool. Section 4 provides an example analysis of a Flight Management System (FMS) task. Section 5 provides conclusions and future work.

The Human-Computer Interaction Process

The conduct of operational procedures and their tasks requires the use of functions (or features) of the automation. To pass the instructions to the automation, the operator interacts with the automation in a cycle that has been described as: goal formulation, execution, and evaluation (e.g. Norman, 1988). Each cycle involves the use of cognitive, motor, and visual activities of an operator (Card, Moran & Newell, 1983).

The performance of the operator in completing a mission task is largely determined by the degree to which the content of the user-interface matches the “semantic space” held by the (domain expert) operator (Kitajima, Blackmon, and Polson, 2002). For example, the “semantic distance” between the way a pilot interprets an Air Traffic Control (ATC) instruction and the availability of a function in the automation to perform the ATC instruction determines the learning and use performance.

Task Analysis Methods

Several techniques have been used to determine the usability of automation (Nielsen, 1992). The most accurate evaluation of the usability of a product is achieved through experimental user testing. Human subjects perform a list of tasks using the automation under test while observers take notes or record the operator’s behavior. The aim is to identify problems on the product or features that users like and are easy to use. Techniques include “think aloud protocols” and eye tracking. Although quantitative data can be collected by measuring time to learn, speed of performance, and rate of human error; this approach is cost prohibitive and can only be conducted at the end of the development cycle when the cost of revisions is highest (Nielsen, 1994).

Alternative approaches that can be used earlier in the life-cycle, fall into two categories: Manual Inspections and Operator Performance Predictions. Manual inspections, such as participatory design (Muller and Kuhn, 1993), cognitive walkthroughs (Wharton, Rieman, Lewis, and Polson, 1994), heuristic evaluations (Nielsen, 1992), and other forms of expert reviews, have been shown to be effective in certain settings (Dumas, 2003) but are subjective and can be biased by group-thinking (Turner & Pratkanis, 1998). These methods also exhibit poor inter-rater reliability (Hetrzum & Jacobsen, 2003) due to differences in granularities of the task definition and the differences in the subjective ratings.

Automated tools, such as CogTool (John, Prevas, Salvucci, and Koedinger, 2004), seek to eliminate these two sources of poor inter-rater reliability by capturing actual end-user button pushes (to eliminate ambiguity in the task definition), and by estimating performance using human performance models such as Keystroke-Level Model (KLM), (Luo & John, 2005). These tools can also be used early in the development cycle.

CogTool provides an easy way to model skilled users' performance behavior through storyboards. To create the storyboards, CogTool users include the sequence of screen shots and specify "hot-spots" or widgets on the screen shots to simulate the user interaction. The screenshots are connected though transitions. Once the screens are connected, the user interacts on the screenshots through the widgets, and CogTool generates an executable script of the actions performed by the user that can be processed by an Operator Performance Model such as KLM (Luo & John, 2005), ACT-R (Anderson et al., 1995) or CORE (Vera, Howes, McCurdy, & Lewis, 2004) to compute a prediction of expert time-on-task.

The HCIPA Method

HCIPA is a manual task/usability analysis inspection method that was designed to address issues with usability in the aviation and space industries (Sherry, et al., 2002, 2006). Specifically, these industries were interested in evaluating usability for measures of the performance of training (e.g. Trials-to-Mastery) and measures of performance of operation (e.g. Probability-to-Complete the Task).

The HCIPA method has its roots in a model of pilot cognition (Polson, Irving, & Irving, 1994) and the Cognitive Walkthrough (Wharton, Rieman, Lewis, Polson, 1994). The HCIPA method, also known as the RAFIV model (Sherry et al., 2002), has two features. First, the HCIPA model decomposes the operator actions to perform a task into six sequential steps. Second, the model assesses the visual salience of prompts to guide the operator in performing each action. Operator actions that are not prompted have been found to result in significantly longer training periods (i.e. Trials-to-Mastery) and increased Probability of Failure-to-Complete a task (Bovair, Kieras, Polson, 1990; Fennel, Sherry & Roberts, 2004; Sherry, Fennell, Feary, Polson, 2006).

The six steps are:

(1) Identify Task: Operator responds to external stimuli to initiate a task. The stimuli may include visual cues (e.g. checklists, labels, prompts and error messages), aural cues (e.g. aural warnings, instruction from personnel), or by remembering to perform a task at a specific time (e.g. prospective memory). This step is a decision-making step.

(2) Select Function: Operator selects the appropriate feature (of function) of the automation. This step is a decision-making step.

(3) Access Correct Display: Operator actions to locate and/or bring-up the correct page or display. In windows-based systems, the access is performed using menus, and tabs. This step requires physical actions to push buttons and select graphical widgets.

(4) Enter: Operator enters data or select parameters to activate the function. This step requires physical actions to type entries, select from lists, etc.

(5) Confirm & Save: Operator checks with co-pilot and Executes (or Saves) the data. This step requires physical actions to push buttons and select graphical widgets.

(6) Monitor: Operator checks the progress of the automation by monitoring the automated commands and status of the system. This step is a decision-making step.

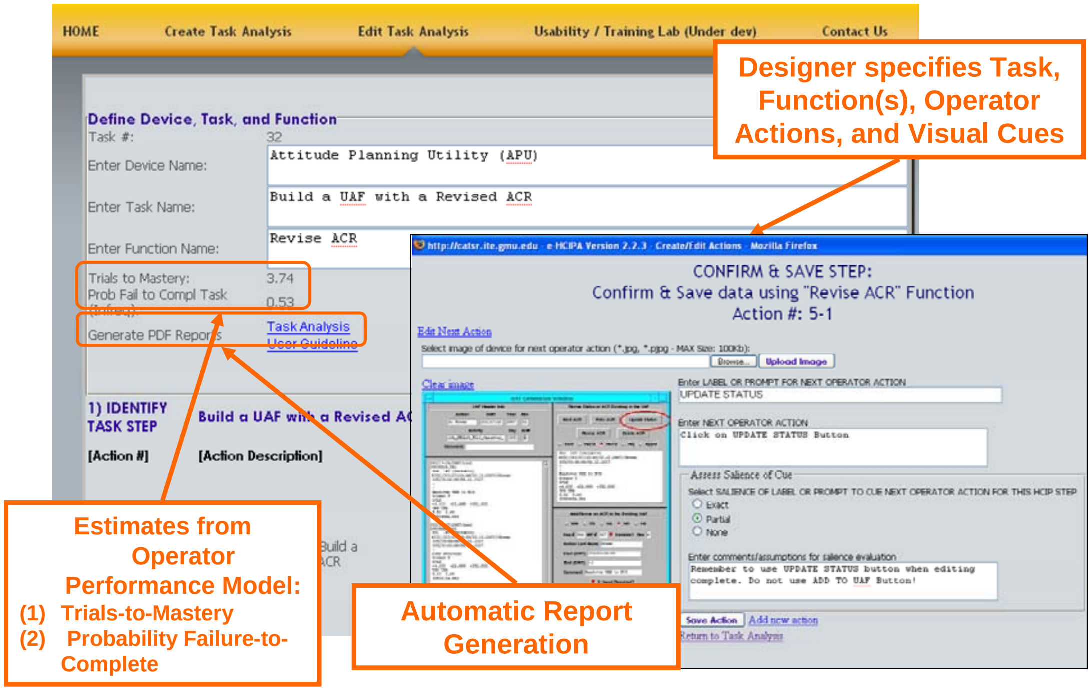
Studies have shown that proficiency in performing the operator actins in each of the steps is reduced significantly, and training costs increased rapidly, when the user interface does not provide salient visual cues to guide the next operator action in each step (Singer & Dekker, 2000). For example, functions that are both hidden in a hierarchy of display pages and are used infrequently, are difficult to learn and (despite advantages in their use) are not used routinely in revenue service operations. Likewise, entries with ambiguous formats that result in the ubiquitous "format error" messages, and accessing functions buried in a confusing navigation hierarchy result in time spent in exploration of the user-interface and trail-and-error. (Kitajima, Blackmon & Polson, 2000).

The HCIPA method formally documents the operator actions in each of the six steps as well as the salience of the visual cues in guiding the next operator action.

The HCIPA method has been successfully applied in several projects across domains (Sherry, Polson & Feary, 2002; Sherry, Fennell, Feary, & Polson, 2006). Theses experiences identified the following issues with the HCIPA method: (1) ambiguity in the granularity of tasks, (2) ambiguity in identification of salient visual cues, (3) issues in reliably assessing the salience of visual cues, (4) no method to compute the Trials-to-Mastery and Probability of Failure to Complete a Task. The tool described in this paper is designed to overcome theses shortfalls and includes an affordable Operator Performance Model to compute Trials-to-Mastery and Probability-of-Failure-to-Complete the task.

The HCIPA Tool

The HCIPA tool provides evaluator: (1) easy way to capture the operator actions required to perform a task in a procedure through a graphical user-interface, (2) the task is stored in a data-base that allows for easy modification and revision, (3) the task model in the data-base enables the execution of human performance models to evaluate the training times and operational efficiency of the task, and (4) the task model in the data-base provides material that can be used for inspections, walkthroughs, and forms the basis of


HOME
Create Task Analysis
Edit Task Analysis
Usability / Training Lab (Under dev)
Contact Us
Designer specifies Task, Function(s), Operator Actions, and Visual Cues
Estimates from Operator Performance Model:
(1) Trials-to-Mastery
(2) Probability Failure-to-Complete
Automatic Report Generation
http://catsr.ite.gmu.edu - e-HCIPA Version 2.2.3 - Create/Edit Actions - Mozilla Firefox
Browse...
Upload Image
Save Action

Define Device, Task, and Function

Task #: 32

Enter Device Name: Attitude Planning Utility (APU)

Enter Task Name: Build a UAF with a Revised ACR

Enter Function Name: Revise ACR

Trials to Mastery: 3.74

Prob Fail to Compl Task (Infreq): 0.53

Generate PDF Reports: Task Analysis, User Guideline

1) IDENTIFY TASK STEP — Build a UAF with a Revised ACR

[Action #] [Action Description]

CONFIRM & SAVE STEP:
Confirm & Save data using "Revise ACR" Function
Action #: 5-1

Edit Next Action

Select image of device for next operator action (*.jpg, *.pjpg - MAX Size: 100Kb):

Clear image

Enter LABEL OR PROMPT FOR NEXT OPERATOR ACTION: UPDATE STATUS

Enter NEXT OPERATOR ACTION: Click on UPDATE STATUS Button

Assess Salience of Cue

Select SALIENCE OF LABEL OR PROMPT TO CUE NEXT OPERATOR ACTION FOR THIS HCIP STEP

○ Exact
● Partial
○ None

Enter comments/assumptions for salience evaluation: Remember to use UPDATE STATUS button when editing complete. Do not use ADD TO UAF Button!

Add new action

Return to Task Analysis

Overview of HCIPA tool
Figure 2.

training material.

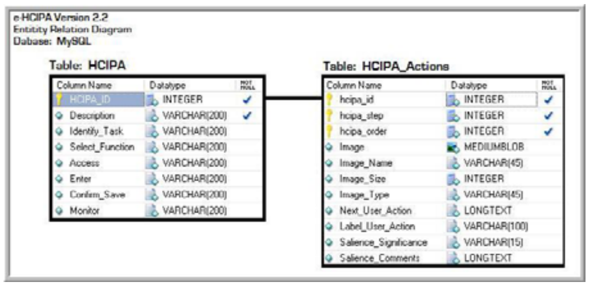
The HCIPA tool is a web based application. The current version of the tool runs on both Internet Explorer and Mozilla Firefox web browsers. The tool requires registration with an e-mail, and is available at no charge. Figure 2 illustrates the user-interface for entry of the tasks. Figure A-1 in the appendix provides a sample report generated from the tool.

Technical Implementation

The HCIPA tool has been developed using PHP 5.2 and MySQL database. Figure 3 shows the Entity-Relationship Diagram of HCIPA.

The database table HCIPA stores the information for the device name, task description and function on fields Description, Identify_Task and Select_Function respectively. Once the user saves a new Task Analysis, the HCIPA tool populates the rest of the fields in the table HCIPA based on the information stored on the fields Identify_Task and Select_Function. Furthermore, two default operator actions are created, one for the Identify_Task step and the other one for Select_Function.

Table HCIPA_Actions stores all operator actions for the given task. The field hcipa_step is a enumerated field that keeps track of current step for the operator action. The values are: 1 for Identify_Task, 2 for Select_Function, 3 for Access, 4 for Enter, 5 for Confirm and Save and 6 for Monitor. The field hcipa_order is a number used to reference the current order of the operator action for the step.



e-HCIPA Version 2.2
Entitity Relation Diagram
Dabase: MySQL

Table: HCIPA

Column Name	Datatype	NOT NULL
HCIPA_ID	INTEGER	✓
Description	VARCHAR(200)	✓
Identify_Task	VARCHAR(200)	
Select_Function	VARCHAR(200)	
Access	VARCHAR(200)	
Enter	VARCHAR(200)	
Confirm_Save	VARCHAR(200)	
Monitor	VARCHAR(200)	

Table: HCIPA_Actions

Column Name	Datatype	NOT NULL
hcipa_id	INTEGER	✓
hcipa_step	INTEGER	✓
hcipa_order	INTEGER	✓
Image	MEDIUMBLOB	
Image_Name	VARCHAR(45)	
Image_Size	INTEGER	
Image_Type	VARCHAR(45)	
Next_User_Action	LONGTEXT	
Label_User_Action	VARCHAR(100)	
Salience_Significance	VARCHAR(15)	
Salience_Comments	LONGTEXT	

Entity-relationship model for HCIPA tool

Figure 3

Case Study

An example analysis of a flightdeck task using the HCIPA method is illustrated in Table A-1 in the Appendix.

Following pre-flight at the gate and entry into the automation the departure runway, the standard instrument departure (SID) and the SID transition, the aircraft is taxi-ing to the runway when the ATC revises the departure clearance with a change in runway (e.g. 16L to 16R). The design of the automation is such that the SID and SID Transition must also be revised along with the revision of the runway. Failure to re-select the SID and SID transition (even though they are displayed as selected) results in the deletion of waypoints in the flightplan route. This is a legitimate design for the automation, but is not visually cued in a salient manner resulting in difficulty in training and in operation.

The first column in Table A-1 lists the six HCIPA steps. The second column lists the operator actions for each step. Column three identifies the label or prompt that is available to guide the operator. Column four shows the salience of the label or prompt to guide the operator to perform the next action for each step. The tool is able to automatically generate several entries. These are shown in italics. The report generated by the tool for this task is included in Appendix A.

The tool calculates the estimated Trials-to-Mastery as 4 (i.e. on average four repetitions of the task is required before the task is mastered and can be performed to proficiency 24 hours later). The Probability of Failure to Complete the Task is estimated as 48% (i.e. 48% of pilots using this feature after a pro-longed absence of more than 3 months will fail to complete the task).

Procedure and Training Mitigation

Based on the HCIPA the procedure and training developers should consider strategies to address this task that exhibits a high Probability of Failure to Complete the task (e.g. > 35%). The procedure may be redesigned to ensure that the Runway, SID and Transition are always modified at the same time, and/or the CRM interaction may include a cross-check of all 3 parameters.

Identify applicable sponsor/s here. *(sponsors)*

In terms of training, the material should include all of the steps and their operator actions. The material should place special emphasis on the "feature" of the automation that requires re-selection of all 3 parameters. Attention should also be brought to all of the operator actions that are not supported by salient visual cues. For example, training materials with GOTCHA highlights to remind pilots "Change one, change all." Also, simulator and proficiency checks should include this task and provide opportunities for repetition.

Conclusions

This paper describes a tool that this intended for use by airline personnel to rapidly evaluate procedures and training material that are being developed for the introduction of new cockpit automation. Specifically, the tool enables evaluators to rapidly and repeatably, assess the Trials-to-Mastery (i.e. time to competence for training) and the Probability of Failure-to-Complete a Task. The tool also provides reports with images of the operator actions. These reports can be archived for certification purposes and/or be used as the basis for training material.

At this time, the tool is available on the web for use at no cost. http://catsr.ite.gmu.edu/HCIPA

Future Work

The tool is continuously under modification based on user's needs and improvements to the model. Currently, research is being conducted in the following areas:

- Operator Performance Models (OPM): Research continues to refine the OPM. Experiments are run with human-in-the-loop subjects performing tasks. Also, affordable cognitive models are being adapted and incorporated into the tools.
- Inter-rater Reliability of the Assessment of the Salience of the Visual Cues: The assessment of the salience of visual cues for prompting the operator's next action is critical for the accuracy of the tool. The current version of the tool relies on the assessment of the salience of the cue by the designer (i.e. None, Partial, Exact). This manual form of assessment suffers several issues. First, the assessment is reliant on the overlap of the designers "semantic state-space" with the end-users "semantic state-space." Recent studies have shown wide variance in semantic state-spaces and large differences between the semantic state-spaces of the designers and end-users. Second, even within a group of end-users and domain experts, the semantic state-space can exhibit a wide distribution.

  This issue will be investigated in two ways. First it is proposed to add a feature of the tool, loosely named, "Usability Lab." This feature will enable the collation of domain experts assessment of the salience of the visual cues. Second, several automated techniques exist to automate the salience assessment. Latent Semantic Analysis, LSA (Landauer & Dumais, 1997; Kitajima, Blackmon, and Polson, 2000) and Scent-based Navigation and Information Foraging in the ACT architecture, SNIF-ACT (Pirolli & Fu, 2003) are two of these automated technique that will be researched to evaluate their feasibility to be included in e-HCIPA.
- Web and data-base upgrades. As new web and data-base technologies become available are wrapped into the tool. This improves the reliability, security, availability, and ease-of-use of the tool.

Acknowledgment

Thank you for technical assistance and suggestions from Peter Polson (University of Colorado), Mike Matessa (Alion Inc.), Karl Fennel (United Airlines). Thank you for support of the research to Steve Young (NASA), Amy Pritchett (NASA). This project was funded by grant from NASA – Aeronautics – Intelligent Integrated Flightdeck program, and by internal George Mason University Foundation Funds.

References

[1] Anderson, J. R., John, B. E., Just, M. A., Carpenter, P. A., Kieras, D. E., & Meyer, D. E. (1995). Production system models of complex cognition. In Proceedings of the Seventeenth Annual Conference of the Cognitive Science Society (pp. 9-12). Hillsdale, NJ: Lawrence Erlbaum Associates

[2] Bovair, S., Kieras, D.E., & Polson, P.G. (1990). The acquisition and performance of text editing skill: A cognitive complexity analysis. Human-Computer Interaction, 5, 1-48.

[3] Card, S., Moran, T. & Newell, A. (1983). The Psychology of Human-Computer Interaction. Hillsdale, NJ: Erlbaum.

[4] Castor-Peck, Steve (2007) Technical Memorandum to NASA Aeronautics – Intelligent Flight-deck

Research Program. NASA-Ames Reseearch Center, Moffet Field,CA.

[5] Dumas, J. S. (2003) User-based evaluations. In J. Jacko and A. Sears (Eds.), The Human-Computer Interaction Handbook. (pp. 1093-1117) Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

[6] Fennell, Karl; Sherry, Lance; Roberts, & Ralph, Jr. (2004). Accessing FMS Functionality The Impact of Design on Learning NASA Technical Report (IH-051; NASA CR-2004-212837).

[7] Hertzum, Morten & Jacobsen, Niels Ebbe (2003): The Evaluator Effect: A Chilling Fact About Usability Evaluation Methods. In International Journal of Human-Computer Interaction, 15 (1) pp. 183-204

[8] John, B., Prevas, K., Salvucci, D. D., & Koedinger, K. (2004). Predictive human performance modeling made easy. In Human Factors in Computing Systems: CHI 2004 Conference Proceedings. New York: ACM Press.

[9] Kitajima, M., Blackmon, M. H., & Polson, P. (2000) G. A Comprehension-based model of Web navigation and its application to Web usability analysis. In People and Computers XIV, Springer, 357--373, 2000.

[10] Landauer, T. K. & Dumais, S. T. (1997) A solution to Plato’s problem: The Latent Semantic Analysis theory of acquisition, induction, and representation of knowledge. Psychological Review, 104, 211–240., 1997.

[11] Luo, L., & John, B. E, (2005) Predicting task execution time on handheld devices using the keystroke-level model, CHI '05 extended abstracts on Human factors in computing systems, April 02-07, 2005, Portland, OR, USA

[12] Muller, Michael J., & Kuhn, S. (1993) “Participatory Design” In: Communications of the ACM special issue on participatory design. 36(4), June 1993.

[13] Nielsen, J. (1992) “The usability engineering life cycle” In Computer Volume 25, Issue 3, Page(s):12 – 22, March 1992.

[14] Nielsen, J. & Landauer, T. K. (1993) A mathematical model of the finding of usability problems, Proceedings of the ACM INTERCHI'93 Conference (Amsterdam, the Netherlands, April 24-29), 206-213.

[15] Nielsen, J, & R.L. Mack (1994) Usability Inspection Methods. John Wiley & Sons, New York, NY, 1994.

[16] Norman, D.A. (1988) "The Design of Everyday Things." MIT Press

[17] Pirolli, P. & Fu, W.-T. (2003). SNIF-ACT: A Model of Information Foraging on the World Wide Web. In Proceedings of the Ninth International Conference on User Modeling, 2003

[18] Polson, P. G., Irving, S., & Irving, J. E. (1994). Final report: Applications of formal methods of human computer interaction to training and the use of the control and display unit. Washington, DC: System Technology Division, ARD 200, Department of Transportation, Federal Aviation Adrninistration.

[19] Sherry, L., Fennell, K., Feary, M., & Polson, P. (2006) Human–Computer Interaction Analysis of Flight Management System Messages. Journal of Aircraft, Vol 43, No. 5, September-October 2006.

[20] Sherry, L., Polson, P. & Feary, M. (2002) Designing User-Interfaces for the Cockpit: Five Common Design Errors, and How to Avoid Them. Paper to be presented at the 2002 SAE World Aviation Congress, Phoenix, AZ (November 5 – 7).

[21] Sherry, L., Polson, P., Feary, M. & Palmer, E. (2002) When Does the MCDU Interface Work Well? Paper presented at the the International Conference on Human-Computer Interaction in Aeronautics, Cambridge, MA (October 22-24).

[22] Singer, G & S.W.A Dekker. (2000) Pilot performance during multiple failures: An empirical study of different warning systems. *Transportation Human Factors*, Volume 2 (1)..pp. 63-76

[23] Turner, M. E., & Pratkanis, A. R. (1998). Twenty-five years of groupthink theory and research: Lessons from the evaluation of a theory. Organizational Behavior & Human Decision Processes, 73(2-3), 105-115.

[24] Vera, A., Howes, A., McCurdy, M., & Lewis, R.L.(2004). A constraint satisfaction approach to predicting skilled interactive performance. Proc. of the SIGCHI Conference on Human Factors in Computing Systems.

[25] Wharton, C., Rieman, J., Lewis, C., & Polson, P. (1994). The cognitive walkthrough method: A practitioner’s guide. In J. Nielsen & R. L. Mack (eds.). Usability inspection methods. New York, NY: John Wiley.

Email Addresses

lsherry@gmu.edu, mmedinam@gmu.edu, michael.s.feary@nasa.gov, john.otiker@wnco.com

Conference Identification

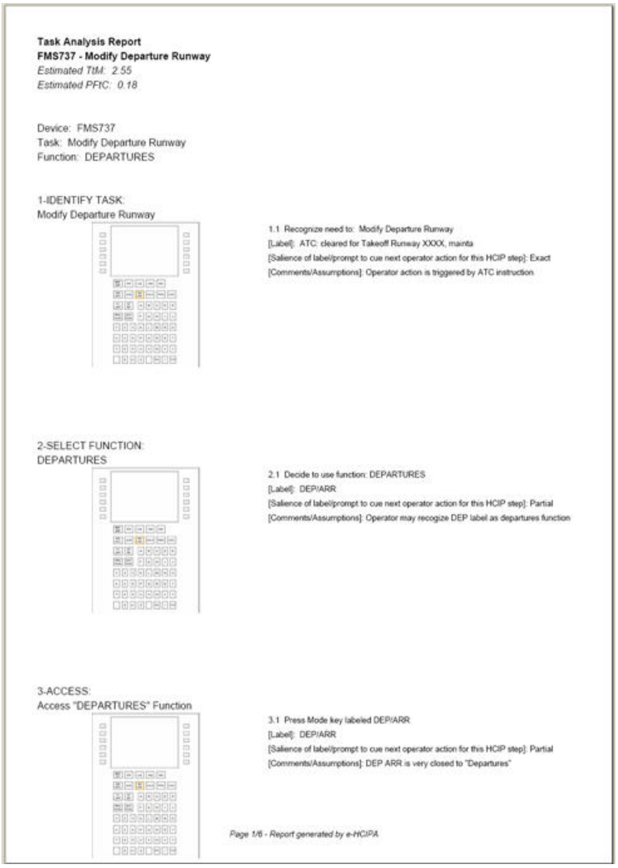
2008 ICNS Conference

5-7 May 2008

Appendix

Task: ATC: ”Cleared Departure 16R, SID FFV6, Trans MLF”			
Step	**Operator Action**	**Label to Prompt Next Operator Action to Achieve Step**	**Salience of Label to Cue Next Operator Action**
Id Task: ATC” Cleared Departure 16R, SID FFV6, Trans MLF”.	*Recognize need to Modify Departure Runway*	ATC instruction	Exact. ATC instruction is unambiguous.
Select Function DEPARTURES	*Decide to use the DEPARTURE Function*	DEP/ARR Mode Key. Also, LEGS and ROUTE Mode Keys.	Partial. The pilot must remember that this task is supported by the DEParture function. This task cannot be executed using LEGS or ROUTE functions.
Access DEPARTURES Function	Press Mode key labeled DEP/ARR	DEP/ARR	Partial. DEP provides partial match with ATC clearance, but competes with LEGS and ROUTE
	Press LS 1L	DEP KSLC	Exact
Enter data for DEPARTURES Function	Line Select XR desired Runway (Press NEXT PAGE or PREV Page Mode Key top see other options)	RUNWAYS	Exact. This is part of the ATC instruction and is displayed.
	Line Select XL desired SID (Press NEXT PAGE or PREV PAGE Mode Key to see other options)	SIDS	None. The correct SID is displayed as Active. There is no reminder to the pilot to reselect the SID.
	Line Select XL desired SID Transition (Press NEXT PAGE or PREV PAGE Mode Key to see other options)	TRANS	None. The SID Transition is only displayed after the SID and Runway have been selected. Also, the correct Transition is displayed as Active
Confirm & Save data using “DEPARTURES” Function	Check SID is <SEL>	SID… <SEL>	Exact. ND shows dashed white line and LEGS page shows changes with MOD
	Check RUNWAY is <SEL>	RUNWAY …<SEL>	Exact. ND shows dashed white line and LEGS page shows changes with MOD
	Check TRANS is <SEL>	TRANS … <SEL>	Exact. ND shows dashed white line and LEGS page shows changes with MOD
	Confirm changes with other pilot	None	Exact. There is no prompt, but standard flightdeck CRM practice to confirm changes.
	Press EXECute Key	EXEC	Exact
Monitor results of "DEPARTURES" Function	Check LEGS includes desired Runway, SID waypoints, and SID Transition waypoints	LEGS	Exact. ND and LEGS pages are updated.
	Check LEGS does not include a – ROUTE DISCONTINUITY – between SID or SID Transition waypoints	None	Partial
	Check ND shows desired course and sequence of waypoints	None	Exact

Table A-1: Example HCIPA analysis of FMS task. The description of the task includes the Steps, Operator Actions, Visual Cues, and an assessment of the salience of the visual cues.



Task Analysis Report
FMS737 - Modify Departure Runway
Estimated TtM: 2:55
Estimated PFtC: 0.18

Device: FMS737
Task: Modify Departure Runway
Function: DEPARTURES

1-IDENTIFY TASK:
Modify Departure Runway

1.1 Recognize need to: Modify Departure Runway
[Label]: ATC: cleared for Takeoff Runway XXXX, mainta
[Salience of label/prompt to cue next operator action for this HCIP step]: Exact
[Comments/Assumptions]: Operator action is triggered by ATC instruction

2-SELECT FUNCTION:
DEPARTURES

2.1 Decide to use function: DEPARTURES
[Label]: DEP/ARR
[Salience of label/prompt to cue next operator action for this HCIP step]: Partial
[Comments/Assumptions]: Operator may recogize DEP label as departures function

3-ACCESS:
Access "DEPARTURES" Function

3.1 Press Mode key labeled DEP/ARR
[Label]: DEP/ARR
[Salience of label/prompt to cue next operator action for this HCIP step]: Partial
[Comments/Assumptions]: DEP ARR is very closed to "Departures"

Page 1/6 - Report generated by e-HCIPA

Sample Report generated by HCIPA tool.
Figure A-2