Evidence-Based Error Analysis: Supporting the Design of Error-Tolerant Systems*

Becky L. Hooey[1], Marco Aurisicchio[2], Robert Bracewell[3], and David C. Foyle[4]

[1] San Jose State University at NASA Ames Research Center, Moffett Field, California
becky.l.hooey@nasa.gov
[2] Department of Mechanical Engineering, Imperial College London
m.aurisicchio@imperial.ac.uk
[3] Rolls-Royce, United Kingdom
rob.bracewell@gmail.com
[4] NASA Ames Research Center, Moffett Field, California
david.c.foyle@nasa.gov

Abstract. This paper proposes an evidence-based process and engineering design tool for linking human error identification taxonomies, and human error prevention and mitigation design principles with the system engineering design process. The process synthesizes the design evidence generated and used during the design and analysis process to clearly demonstrate that credible error threats have been identified and considered appropriately in the design of the system. In doing so, it supports the designer in managing design solutions across the entire design process, leaves a design trace that is transparent and auditable by other designers, managers, or certification experts, and manages the complex interactions among other systems and sub-systems.

Keywords: error-tolerant design, human error, design rationale, designVUE.

1 Introduction

Error-tolerant systems are systems that are robust to human error, in that they guard against errors occurring whenever possible, and support efficient detection and recovery of errors when they do occur [1, 2]. The need for error-tolerant systems has long been recognized and applies to safety-critical systems such as in the aviation, space, medical, and nuclear domains, as well as to the design of ‘everyday things’ such as automated teller machines, consumer electronics, and home appliances [3].

1.1 Requirements for Error-Tolerant Design of Flight Deck Avionics

In 2013, The United States Federal Aviation Administration (FAA) adopted a new regulation (14 CFR 25.1302) that amends design requirements in the airworthiness standards for transport category airplanes to minimize the occurrence of design-related

* The rights of this work are transferred to the extent transferable according to title 17 U.S.C. 105.

M. Kurosu (Ed.): Human-Computer Interaction, Part III, HCII 2014, LNCS 8512, pp. 401–412, 2014.

flightcrew errors and to better enable a flightcrew member to detect and manage errors when errors do occur [4]. In harmony with existing European Aviation Safety Agency (EASA) regulations [5], this regulation recognizes that since flightcrew errors will occur, even with a well-trained and proficient flightcrew operating well-designed systems, system and equipment design must support management of those errors to avoid safety consequences. To the extent practicable, installed equipment must incorporate means to enable the flightcrew to manage errors resulting from flightcrew interactions with the equipment that can be reasonably expected to occur in service. The FAA Advisory Circular (AC 25.1302-1) specifies that certification will likely require multiple forms of compliance (including Statement of Similarity, Design Description, Calculation/Analysis, Evaluations, and Test) and calls for means of compliance that are methodical and complementary to, and separate and distinct from, airplane system analysis methods such as system safety assessments [4].

1.2 The Challenge of Designing Error-Tolerant Systems

Demonstrating error-tolerance of any system poses challenges for designers. Given that errors will occur, even with well-trained operators and well-designed equipment, demonstrating error-free performance in simulation or operational tests is an unreasonable goal. However, if attained, it simply shows that the specific confluence of variables tested did not combine to create an error during the observation period. Even more challenging, is demonstrating adequate design solutions that support error detection and consequence mitigation. This requires a systematic approach to comprehensively identify all potential errors and link them to related design strategies. This quickly becomes intractable when integrated across the entire design lifecycle of a complex system. Not only is it difficult for designers to ensure they have systematically and comprehensively addressed all potential for human error, it is even more difficult to demonstrate the error tolerance of the systems to outside observers.

Evidence-based safety arguments, such as Safety Cases or Assurance Cases have been gaining support as a method to demonstrate that all critical hazards have been eliminated or adequately mitigated in safety-critical systems [6]. A Safety Case is a comprehensive safety argument that communicates how evidence generated from testing, analyses, and review, collectively satisfies claims concerning safety [7]. It aims to make the rationale connecting the design process to the hazard analyses explicit [6], thus *enabling* reviewers and project engineers to understand why a mitigation is effective, see the supporting evidence, and have more confidence in the behavior of the system during operations [8].

1.3 Objective

While Safety Cases aim to identify all potential hazards, their focus tends to be on hazards related to software and hardware, and less so on human error. This paper proposes an evidence-based approach for linking human error identification taxonomies, and human error prevention and reduction design principles with the systems engineering design process. The approach synthesizes design evidence to ensure that credible error threats have been identified and considered appropriately in the design of the system. In doing so, it aims to support the designer in managing design solutions across the entire design process, and manages the complex interactions among other systems and sub-systems, while leaving a design trace that is transparent and auditable by other designers, managers, or certification experts.

2 Evidence-Based Approach for Error-Tolerant Design

The proposed evidence-based approach (see Figure 1) links Human Error Identification analyses and Human Error Mitigation techniques with design evidence (the design decisions and rationale generated during the design process) using an established engineering design knowledge capture tool known as designVUE [9, 10].

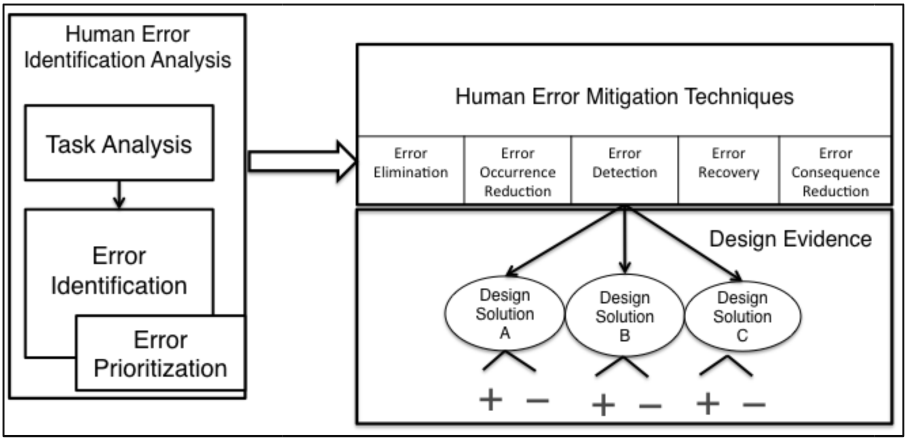


Fig. 1. Evidence-based approach for error-tolerant design

It supports the traceability of evidence-based rationale related to error identification, error prioritization, and error mitigation through design. The three components of this evidence-based approach shown in Figure 1 are described individually in sections 2.1, 2.2, and 2.3. The integration of the three is described using a case study of the design of a flight deck Data Communication (DataComm) system (Section 3.0).

2.1 Human Error Identification Analysis

Many human error identification taxonomies are available in the literature including: Systematic Human Error Reduction and Prediction Approach (SHERPA) [11], Human Error Hazard and Operability Study (HAZOP) [12], Human Error In Systems Tools (HEIST) [13] and Human Error Template (HET) [14]. See also [15] for description and examples of these and other human error identification taxonomies. The structure and dimension of the actual error taxonomy adopted depends on the system being developed as well as on the aim of the analysis [16].

In this case study, we adopted the Human Error Template (HET), as it was designed specifically for application on the aircraft flight deck. HET is a checklist that includes 12 potential error modes that were selected based on a study of actual pilot error incidence and existing error modes in contemporary human error identification methods. The HET is applied to each bottom-level task step in a hierarchical task analysis. The analyst indicates which of the HET error modes are credible (if any) for each task step. For each credible error, the analyst provides a description of the form that the error would take. The analyst then determines the outcome or consequence associated with the error. Finally, the analyst estimates the likelihood of the error (Low, Medium, High) and the criticality of the error (Low, Medium, High). If the error is given a high rating for both likelihood and criticality, the aspect of the system involved in the task step is then rated as a 'fail'.

2.2 Human Error Mitigation Techniques

The choice of suitable human error mitigation techniques depends on the system and domain under study. A large emphasis in the literature has focused on the development of design guidelines. Examples of guidelines include design strategies such as forcing functions including interlocks, lockins, and lockouts [3] and guidelines for coping with human errors through system design, including errors related to learning processes, interference among control structures, lack of resources, and stochastic errors [16]. Design techniques to avoid human error consequences in nuclear plant operations and maintenance are provided by [17]. In addition to providing guiding principles for addressing human errors (e.g., make goals and system state visible, provide a good conceptual model, make the acceptable regions of operation visible, etc.), they also provide error management strategies for the following: 1) Eliminate error occurrence; 2) Reduce error occurrence; 3) Eliminate error consequences, which is further subdivided into error detection, error recovery, and consequence prevention; and, 4) Reduce error consequence. This human error mitigation taxonomy is adapted as shown in Figure 1 and Section 3.0.

2.3 Design Evidence

Evidence-based design rationale is a representation of the reasoning behind the design of an artifact [18]. Design rationale, includes the reasons behind a design decision, the justification for it, the other alternatives considered, the tradeoffs evaluated, and the argumentation that led to the decision [19]. Sources of evidence-based rationale may range from anecdotal descriptions – either substantiated or not, to detailed data derived from analyses, experiments, or operational tests [18]. However, this information is rarely captured in a systematic and usable format because there are few tools that adequately facilitate and support the capture of these critical decisions.

One exception is a tool called designVUE [9,10], an Issue Based Information System (IBIS) derivative [20] developed for the purpose of capturing, structuring, and analyzing design decisions as they are proposed throughout the design process. Using the evolution of the IBIS notation shown in Figure 2, designVUE allows one to build directed graphs, where nodes representing issues to be resolved, alternative answers, and arguments in favor (pro) and against (con), are linked by arcs. For each issue and answer, the status can be indicated as an open answer; accepted answer; likely answer; unlikely answer; or rejected answer (see Figure 2).

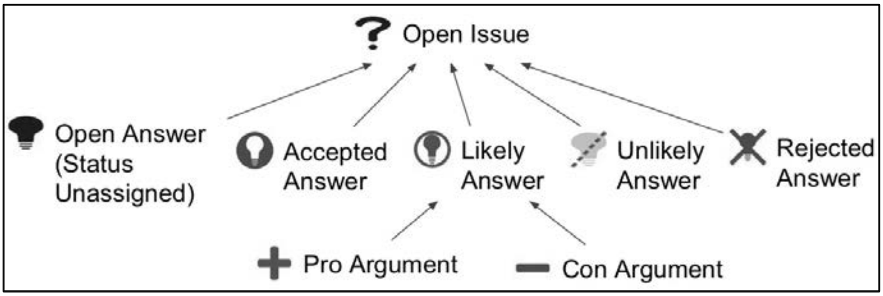


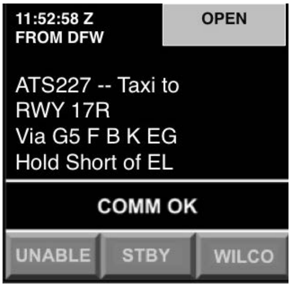
Fig. 2. IBIS notation as instantiated in designVUE

A graph generated in designVUE is captured and saved in a single file and its nodes can be linked to those of graphs in other files through a bi-directional hyperlink called a *wormhole*. In addition, the tool supports a mono-directional hyperlink to web resources as well as to files in local and shared folders. The designVUE tool was inspired by, and builds on, successful application of the Design Rationale editor (DRed) tool used by Rolls-Royce to support the capture of design rationale [21] and integrated information spaces covering product planning, specification, design and service [22].

In the case study that follows, designVUE is applied to the design of error-tolerant systems by linking human error mitigation design solutions to the human error identification analysis. It provides a context-rich digital design book that documents and links the errors that were considered and the decisions adopted to either eliminate or minimize their occurrence or mitigate the consequences.

3 Case Study: Flight Deck Data Communication (DataComm) System for NextGen Surface Operations

A case study was created for the purpose of demonstrating the design of an error-tolerant system – specifically, a flight deck Data Communication (DataComm) system used by pilots to receive and respond to Air Traffic Control taxi clearances (see Figure 3, [23]). DataComm is akin to receiving a text message from Air Traffic Control (ATC). Most simply, a taxi clearance is a single text message that lists the taxiways that the pilot must follow and the destination. For example, in Figure 3, Runway 17R is the assigned departure runway and G5, F, B, K and EG are taxiway identifiers for the assigned taxi route. A taxi clearance may include a requirement to hold short of a specified taxiway (e.g., HOLD SHORT of EL, see Figure 3). Pilots are required to indicate if they 'Will Comply' (WILCO) or if they are 'Unable to Comply' (UNABLE).



11:52:58 Z
FROM DFW

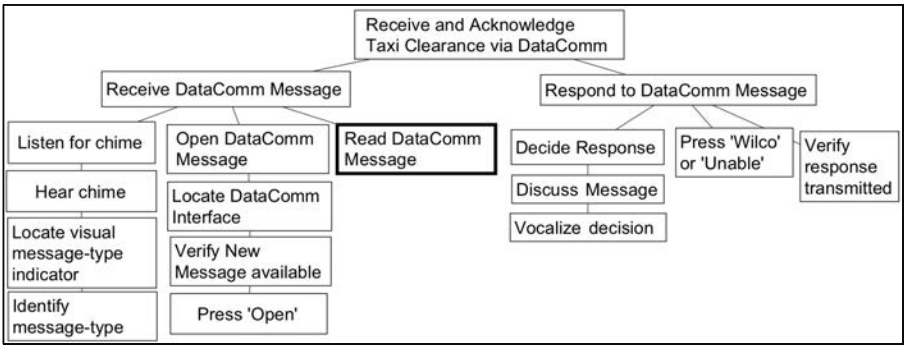
ATS227 -- Taxi to
RWY 17R
Via G5 F B K EG
Hold Short of EL

COMM OK

Fig. 3. DataComm System

3.1 Human Error Identification Analysis

Consistent with the HET process, the human error identification analysis begins with a hierarchical task analysis to identify the bottom-level tasks for further analysis. The task analysis for receiving and responding to a taxi clearance via DataComm is shown below in Figure 4. The human error identification analysis can be completed on each low-level subtask. In designVUE, each task box can be bi-directionally linked to the subsequent error-analyses. The case study that follows analyzes the low-level task 'Read DataComm Message'.



Receive and Acknowledge Taxi Clearance via DataComm
Receive DataComm Message
Respond to DataComm Message
Listen for chime
Hear chime
Locate visual message-type indicator
Identify message-type
Open DataComm Message
Locate DataComm Interface
Verify New Message available
Press 'Open'
Read DataComm Message
Decide Response
Discuss Message
Vocalize decision
Press 'Wilco' or 'Unable'
Verify response transmitted

Fig. 4. Hierarchical Task Analysis

To ensure systematic and comprehensive consideration of potential errors, the HET taxonomy was implemented in designVUE as shown in Figure 5. As the starting point for the analysis, the question 'What errors could occur?' and the 12 possible HET error categories are provided in open status (neither accepted nor rejected).

Each of the 12 potential error categories were assigned a status to indicate that they are either credible (indicated by a green light bulb icon) or not-credible (indicted by a red light bulb icon with an X), as shown in Figure 6. Each credible error was further broken down into sub-error classifications indicating all potential error manifestations. For example, in Figure 6, the 'Task Execution Incomplete' error could

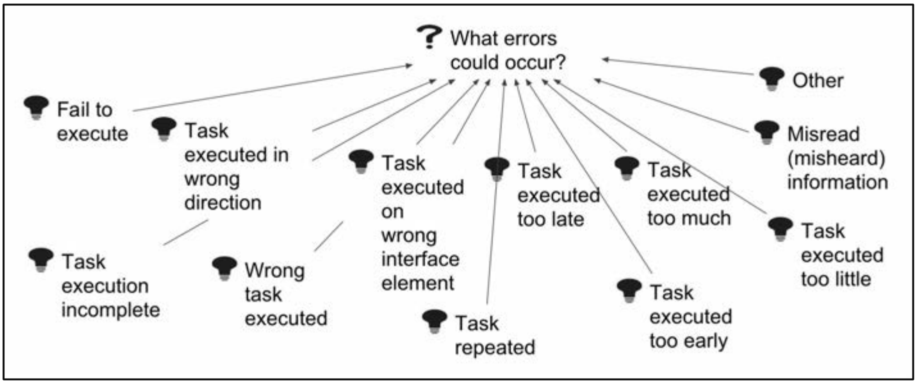


Fig. 5. Human-error Template (HET) taxonomy in designVUE

manifest such that the pilot may begin to read the taxi clearance, but not complete the entire message and miss the hold instruction. This offers the first level of traceability allowing an outside auditor to independently assess the validity of the errors selected for further design consideration.

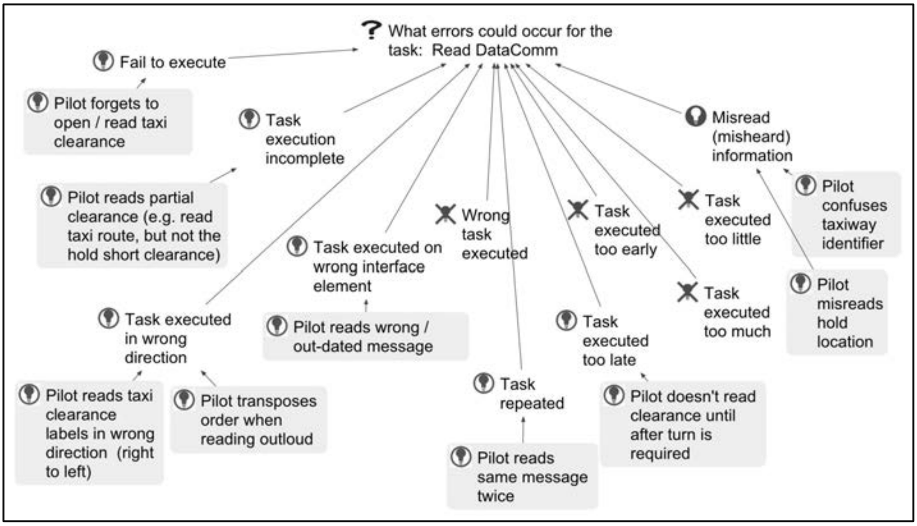

What errors could occur for the task: Read DataComm
Fail to execute
Pilot forgets to open / read taxi clearance
Task execution incomplete
Pilot reads partial clearance (e.g. read taxi route, but not the hold short clearance)
Task executed on wrong interface element
Pilot reads wrong / out-dated message
Wrong task executed
Task executed too early
Task executed too little
Task executed too much
Misread (misheard) information
Pilot confuses taxiway identifier
Pilot misreads hold location
Task executed in wrong direction
Pilot reads taxi clearance labels in wrong direction (right to left)
Pilot transposes order when reading outloud
Task repeated
Pilot reads same message twice
Task executed too late
Pilot doesn't read clearance until after turn is required

Fig. 6. Human-Error Identification for 'Read DataComm' Task

Evidence that supports the credibility of each error category can also be provided. Figure 7 shows just one branch of the Human Error Identification analysis. The error type Task Execution Incomplete is expanded to depict five arguments that justify its classification as a credible error. In this example, evidence took the form of observations of events that lead the pilots to err by reading only part of the taxi clearance during a pilot-in-the-loop simulation. Evidence for rejecting non-credible errors can also be provided to document why an error was not selected for further analysis (not shown).

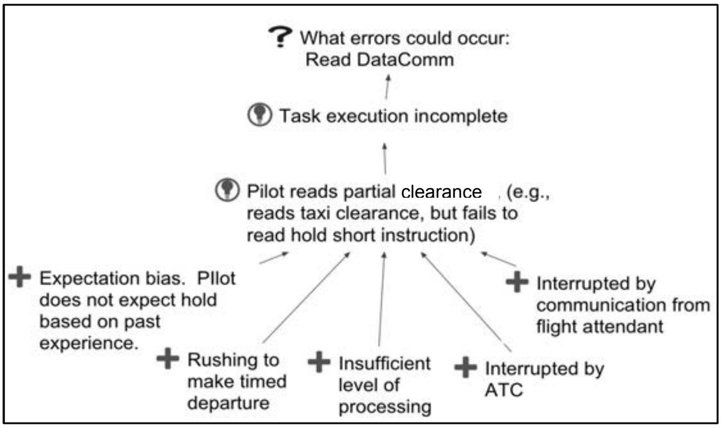

? What errors could occur: Read DataComm
Task execution incomplete
Pilot reads partial clearance (e.g., reads taxi clearance, but fails to read hold short instruction)
+ Expectation bias. PIlot does not expect hold based on past experience.
+ Rushing to make timed departure
+ Insufficient level of processing
+ Interrupted by ATC
+ Interrupted by communication from flight attendant

Fig. 7. Evidence to Support Error Credibility

3.2 Error Prioritization

The credible error threats were then each rated by Subject Matter Experts (SMEs) for likelihood (low, medium, high) and criticality of consequences (low, medium, high). Using the HET criteria, those that scored 'high' on both scales were selected for further analyses. Evidence to support the ratings was also captured (see Figure 8). Evidence may be a subjective assessment by domain experts, or a more objective, quantitative analysis of error likelihood. Because both the ratings and the evidence are made explicit in this transparent design process, the ratings can be revisited with a more informed perspective as the design proceeds.

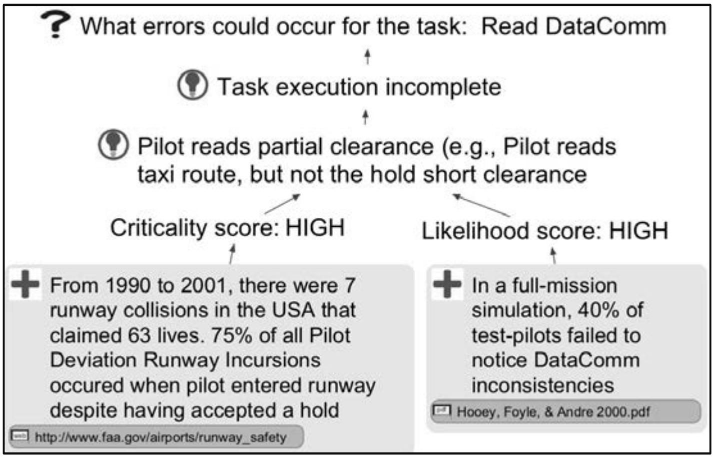

? What errors could occur for the task: Read DataComm
Task execution incomplete
Pilot reads partial clearance (e.g., Pilot reads taxi route, but not the hold short clearance
Criticality score: HIGH
Likelihood score: HIGH
+ From 1990 to 2001, there were 7 runway collisions in the USA that claimed 63 lives. 75% of all Pilot Deviation Runway Incursions occured when pilot entered runway despite having accepted a hold
http://www.faa.gov/airports/runway_safety
+ In a full-mission simulation, 40% of test-pilots failed to notice DataComm inconsistencies
Hooey, Foyle, & Andre 2000.pdf

Fig. 8. Criticality and Likelihood Scores with Evidence

3.3 Design Evidence

Each prioritized error was then linked to design evidence that documents how the designers took steps to prevent or mitigate the error. A taxonomy of error prevention and mitigation adapted from [17] was implemented that includes design techniques to: 1) eliminate error; 2) reduce occurrence of the error; 3) aid error detection; 4) aid error recovery; and, 5) minimize error consequences. While designers can replace this with their own taxonomy, the use of a taxonomy serves to ensure systematic consideration of all error prevention and mitigation strategies. Typically, more than one error solution may be required. This process is demonstrated by showing the design trace for how designers incorporated features to reduce the occurrence of, and aid detection of, the 'Incomplete Task Execution' error in which the taxi clearance is read, but the hold short instruction is missed.

Error Reduction Design Evidence. Figure 9 depicts the design considerations associated with reducing the occurrence of the Task Execution Incomplete error. Four design solutions were considered to maximize the salience of the hold short taxi instruction: AllCaps, Reverse Video, Color-coding, and Blinking/Flashing. During design delibrations, 'pro' and 'con' arguments were provided to either support or refute each design solution in the form of data from simulations and tests, industry or government standards, design guidelines, or argumentation from design team members or domain experts. Hyperlinks were made to further tie the rationale statement to simulation reports or other documents spreadsheets and web URLs.

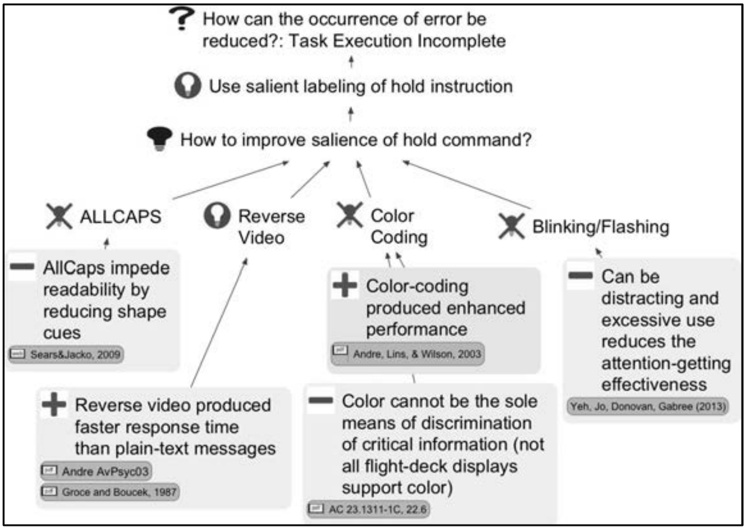

How can the occurrence of error be reduced?: Task Execution Incomplete
Use salient labeling of hold instruction
How to improve salience of hold command?
ALLCAPS
Reverse Video
Color Coding
Blinking/Flashing
AllCaps impede readability by reducing shape cues
Sears&Jacko, 2009
Color-coding produced enhanced performance
Andre, Lins, & Wilson, 2003
Can be distracting and excessive use reduces the attention-getting effectiveness
Yeh, Jo, Donovan, Gabree (2013)
Reverse video produced faster response time than plain-text messages
Andre AvPsyc03
Groce and Boucek, 1987
Color cannot be the sole means of discrimination of critical information (not all flight-deck displays support color)
AC 23.1311-1C, 22.6

Fig. 9. Error Reduction Design Evidence

As shown in Figure 9, the use of reverse video to increase salience of the hold command was selected and supported by empirical data that showed that reverse video produced faster response time than plain, unformatted, text. A hyperlink to the document containing the simulation data is embedded. The other solutions were each refuted with evidence from empirical evaluations or flight deck design guidelines. 'Color-coding' was rejected because despite showing enhanced performance, it was noted that not all aircraft are equipped with DataComm equipage that supports the use of color embedded in DataComm text. Should this hardware constraint be removed in the future, the design solution can be reconsidered without recreating the original rationale.

Error Detection Design Evidence. Figure 10 presents the design evidence associated with aiding the detection of the error: Task Execution Incomplete. That is, if the pilot did fail to read the hold short command in the DataComm message, what features can be implemented to support the pilots' ability to detect the error before they reached their hold location? Two categories of design solutions were considered: Information Redundancy and Procedures.

The design solution, 'Information Redundancy" refers to the designers' recommendation that the hold short information embedded in the text DataComm message should also be presented redundantly, and in a graphical form, on the pilots Navigation Display. In designVUE, a graphical prototype of the design concept and hyperlinks to two empirical study reports that have tested a similar concept are embedded. This demonstrates how designVUE enables linkages across systems and sub-systems enabling information traceability. This error-mitigating design solution involves a different piece of flight deck equipment. Linking the design requirement of one to the design solution of the other reduces the risk that the graphical hold feature may be omitted from the Navigation Display leaving the DataComm system vulnerable to error. Assume, for example, that following the proposal to introduce the flight deck Navigation Display a new team is tasked to develop it. If this team captures and deliberates the system requirements in another designVUE file, the root of the requirement graph can be bi-directionally hyperlinked to the answer node in the file in Figure 10 where the solution was initially conceived. Should the team decide to capture the requirements in a spreadsheet or text document these can still be hyperlinked to the answer node in Figure 10.

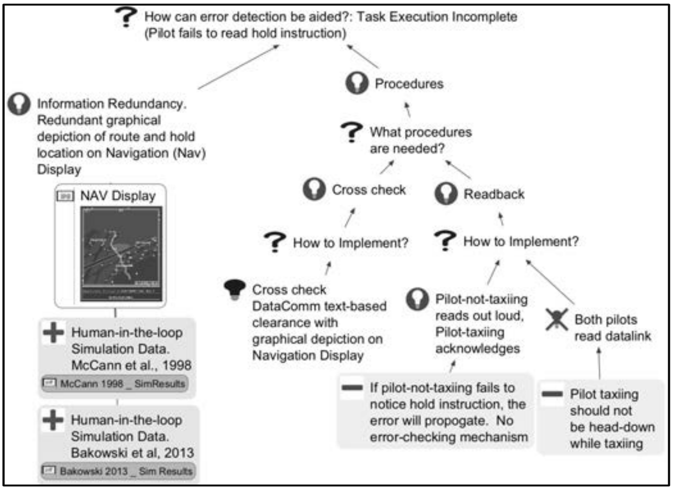

How can error detection be aided?: Task Execution Incomplete (Pilot fails to read hold instruction)
Information Redundancy. Redundant graphical depiction of route and hold location on Navigation (Nav) Display
NAV Display
Human-in-the-loop Simulation Data. McCann et al., 1998
McCann 1998 _ SimResults
Human-in-the-loop Simulation Data. Bakowski et al, 2013
Bakowski 2013 _ Sim Results
Procedures
What procedures are needed?
Cross check
Readback
How to Implement?
How to Implement?
Cross check DataComm text-based clearance with graphical depiction on Navigation Display
Pilot-not-taxiing reads out loud, Pilot-taxiing acknowledges
Both pilots read datalink
If pilot-not-taxiing fails to notice hold instruction, the error will propogate. No error-checking mechanism
Pilot taxiing should not be head-down while taxiing

Fig. 10. Error Detection Design Evidence

Also of note is the ability to document procedural solutions (see Figure 10, right branch). Managing human error in complex systems, such as avionics, is a multi-faceted problem that includes not only physical design of multiple interacting systems, but also operations, procedures, training, and maintenance. Some [e.g., 24] have advocated for tight integration between physical design and the design of operational procedures to guard against unanticipated interactions between the procedures and the system (e.g., when the system enables tasks that are unauthorized by procedures; when information supplied by the device does not agree with information provided through procedures; or when a system is designed assuming a set of procedures, which are later changed, or vice versa). The assumptions made about procedures by the design team can be archived here, and are available for review as procedures and operations are developed and iterated in parallel with equipment design.

4 Discussion

In this paper, a process for systematically addressing and managing error in the design of complex systems was proposed. The process linked human error identification analyses and human error mitigation strategies to the system engineering design process. It supported design rationale capture for each decision using a semi-formal modeling technique. In doing so, the treatment of human error is inserted into the design process, in a manner that enables transparency, and supports integration across sub-systems, operations, and procedures. The result is a visual design logbook that synthesizes the design evidence generated and used during the design and analysis process to clearly demonstrate that credible error threats have been identified and considered appropriately in the design of the system.

Acknowledgments. This research was funded by the National Aeronautics and Space Administration (NASA) Aviation Safety Program, (System-wide Safety Assurance: Human Systems Solutions) and the United Kingdom Engineering and Physical Sciences Research Council (EPSRC) through the Pathways to Impact funding scheme.

References

1. Rasmussen, J.: Skills, Rules, and Knowledge: Signals, Signs and Symbols and Other Distinctions in Human Performance Models. IEEE Transactions: SMC-13, 257–267 (1983)
2. Rouse, W.B., Rouse, S.H.: Analysis and Classification of Human Error. IEEE Transactions: SMC-13, 539–549 (1983)
3. Norman, D.: The Design of Everyday Things. Basic Books, New York (2002)
4. United States Federal Aviation Administration, http://www.faa.gov/documentLibrary/media/Advisory_Circular/AC_25.1302-1.pdf
5. European Aviation Safety Agency, http://www.easa.europa.eu/agency-measures/docs/certification-specifications/CS-25/CS-25%20Amdt%2012.pdf

6. Denney, E., Pai, G.: A Lightweight Methodology for Safety Case Assembly. In: Ortmeier, F., Lipaczewski, M. (eds.) SAFECOMP 2012. LNCS, vol. 7612, pp. 1–12. Springer, Heidelberg (2012)
7. Denney, E.W., Pai, G.J., Habli, I.: Perspectives on Software Safety Case Development for Unmanned Aircraft. In: 42nd Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN) (June 2012)
8. Goodenough, J.B., Barry, M.R.: Evaluating Hazard Mitigations with Dependability Cases. White Paper (April 2009), http://www.sei.cmu.edu/library/abstracts/whitepapers/dependabilitycase_hazardmitigation.cfm/
9. Baroni, P., Romano, M., Toni, F., Aurisicchio, M., Bertanza, G.: An Argumentation-based Approach for Automatic Evaluation of Design Debates. In: Leite, J., Son, T.C., Torroni, P., van der Torre, L., Woltran, S. (eds.) CLIMA XIV 2013. LNCS, vol. 8143, pp. 340–356. Springer, Heidelberg (2013)
10. designVUE, http://www3.imperial.ac.uk/designengineering/tools/designvue/
11. Rasmussen, J., Vicente, K.J.: Coping with Human Errors through System Design: Implications for Ecological Interface Design. Int. J. Man-Mach. Stud. 31, 517–534 (1989)
12. Embrey, D.E.: SHERPA: A Systematic Human Error Reduction and Prediction Approach. In: International Meeting on Advances in Nuclear Power Systems, Knoxville, Tennessee (1986)
13. Neogy, P., Hanson, A.L., Davis, P.R., Fenstermacher, T.E.: Hazard and Barrier Analysis Guidance Document. Department of Energy, Office of Operating Experience Analysis and Feedback, Report No. EH-33 (1996)
14. Reinach, S., Viale, A., Green, D.: Human Error Investigation Software Tool (HEIST). Final Report (2007)
15. Marshall, A., Stanton, N., Young, M., Salmon, P., Harris, D., Demagalski, J., Waldmann, T., Dekker, S.: Development of the Human Error Template – A New Methodology for Assessing Design Induced Errors on Aircraft Flight Decks. Final Report (2003)
16. Salmon, P., Stanton, N.A., Walker, G.: Human Factors Design Methods Review (2003). HFIDTC/WP1.3.2/1 v 1 (2003)
17. Chen-Wing, S.L.N., Davey, E.C.: Designing to Avoid Human Error Consequences. In: Second Workshop on Human Error, Safety, and System Development, Seattle (1998)
18. Buckingham Shum, S., Hammond, N.: Argumentation-based Design Rationale: What Use at What Cost? Int. J. Hum-Comput. St. 40(4), 603–652 (1994)
19. Lee, J.: Design Rationale Systems: Understanding the Issues. IEEE Expert 12(3), 78–85 (1997)
20. Kunz, W., Rittel, H.: Issues as Elements of Information Systems, Working Paper 131, Inst. Urban and Regional Dept., Univ. Calif. at Berkeley (1970)
21. Bracewell, R.H., Wallace, K., Moss, M., Knott, D.: Capturing Design Rationale. Computer Aided Design 41(3), 173–186 (2009)
22. Aurisicchio, M., Bracewell, R.H.: Capturing an Integrated Design Information Space with a Diagram Based Approach. J. Eng. Design 24(6), 397–428 (2013)
23. Bakowski, D.L., Hooey, B.L., Foyle, D.C., Wolter, C.A., Cheng, L.W.S.: NextGen Flight Deck Surface Trajectory-based Operations (STBO): Contingency Holds. In: 32nd Digit Avion Syst. Con., Syracuse (2013)
24. Pritchett, A.: Simultaneous Design of Cockpit Display of Traffic Information and Air Traffic Management Procedures. In: 17th Digit Avion Syst. Con., pp. 36/1–36/9. AIAA (1998)