Human Interface/ Human Error

18-849b Dependable Embedded Systems

Charles P. Shelton

February 25, 1999

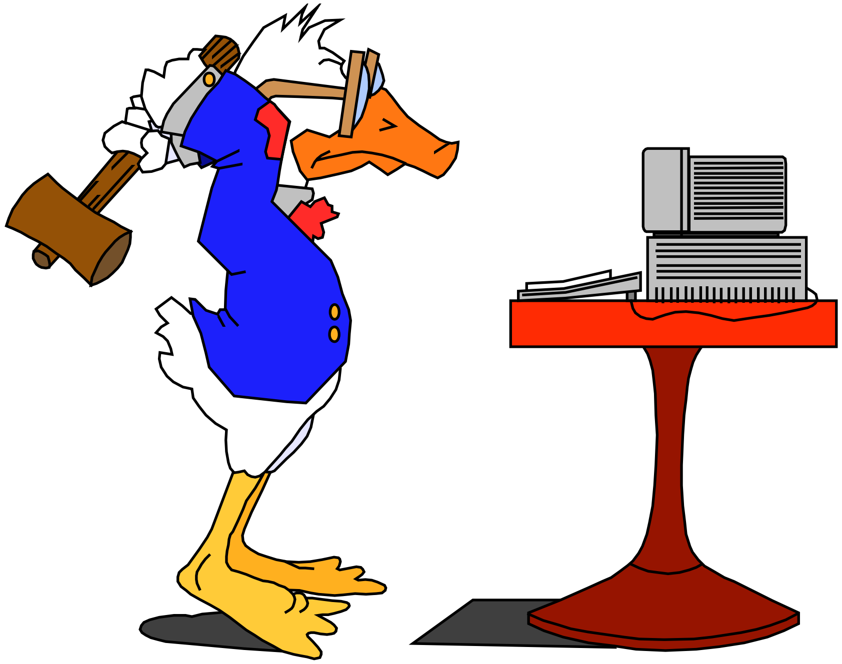
Required Reading: **Murphy, Niall;** ***Safe Systems Through Better User Interfaces***
Supplemental Reading: **Burns, A.;** ***The HCI component of dependable real-time systems***

Authoritative Books: **Reason, James;** ***Human Error***
Nielsen, Jacob; ***Usability Engineering***

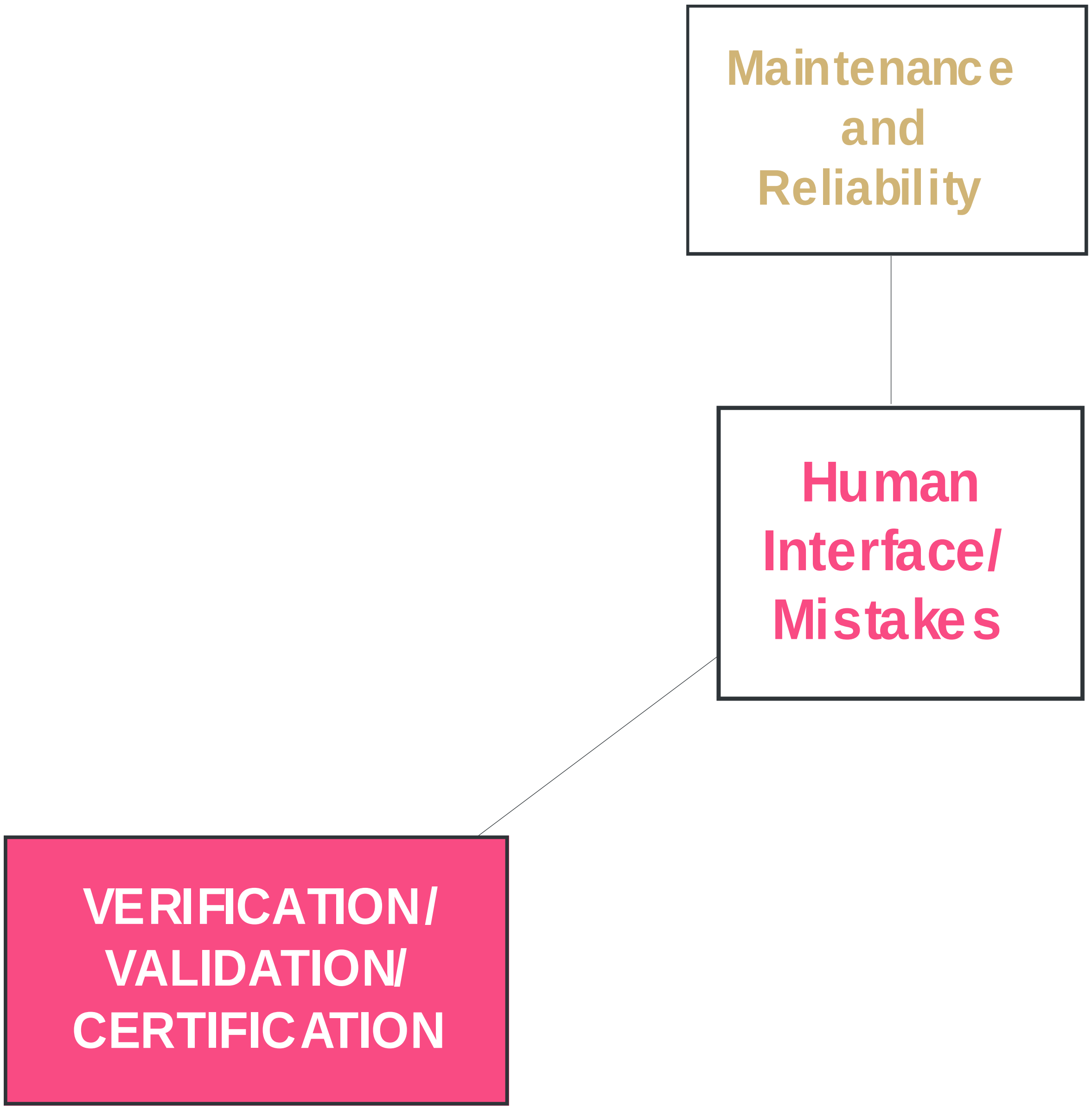
Carnegie Mellon

Overview: Human Interface/Human Error

- **Introduction**
- **Key concepts**
  - Sources of Human Error
  - HCI problems
  - Usability versus Safety
- **Techniques for User Interface Evaluation**
  - Inspection Methods
  - Empirical Methods
- **Relationship to other topics**
- **Conclusions & future work**

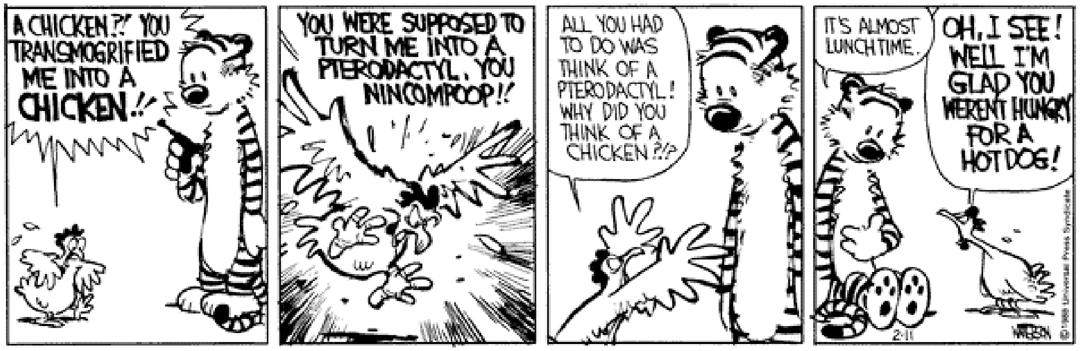


YOU ARE HERE


Maintenance and Reliability
Human Interface/ Mistakes
VERIFICATION/ VALIDATION/ CERTIFICATION

Introduction

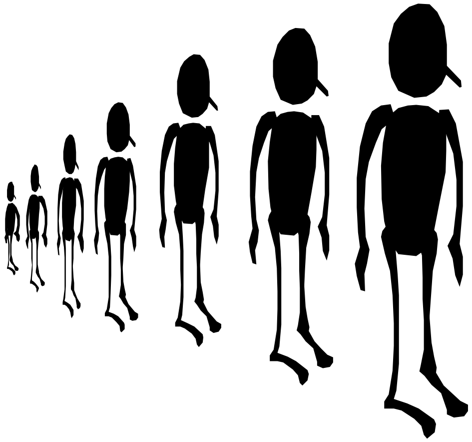
- **“Human error” is the source of most problems in any embedded system**



- **System Design Errors**
  - Incomplete specifications
  - Design defects
  - Implementation errors (bugs in software, manufacturing defects)
- **Errors from the operator or Human-Computer Interface (HCI)**
  - Poorly designed user interface
  - Operator error
  - Maintenance errors
  - Designing HCI to provide appropriate response and feedback for the operator and minimize and compensate for operator error

Sources of Human Error

- **Passive humans are the failsafe when errors occur**
  - Operator is removed from control of the system, but expected to prevent system breakdowns
  - People have short attention spans and will adapt to common case
  - If system usually works without their intervention, they will be slow to react to exceptional conditions
- **Humans make more mistakes under stressful conditions**
  - But are the only part of the system capable of dealing with truly exceptional conditions
- **Repetitive tasks encourage mistakes**
  - When you perform a task you've done a hundred times before, you don't pay attention and will tend to make more mistakes



Stress Factors

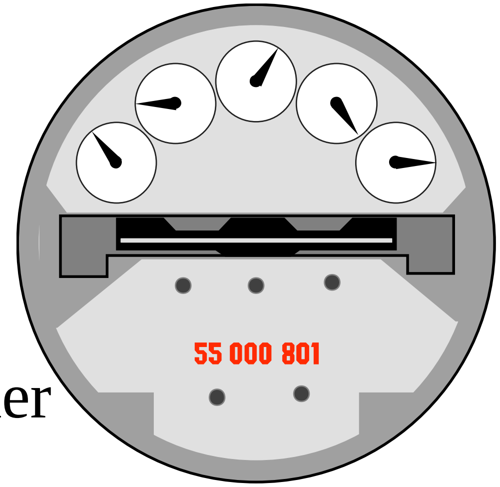
- **Human performance will degrade as stress levels increase**
- **Factors contributing to stress:**
  - Unfamiliar situations/exceptional conditions
  - Perceived level of threat/danger
  - Time constraints
- **Training can help**
  - Rigorous training can make exceptional conditions feel routine and reduce stress
  - However, training cannot completely compensate for anxiety in unique unanticipated situations

Human Error Probabilities

Error Type	Human-error Probability
Errors for very high stress levels	0.3
Fails to act correctly in first 30 minutes of a stressful situation	0.l
Fails to act correctly after first few hours in a high-stress scenario	0.01
Human-performance limit: single-operator	0.0001

HCI Problems

- **Information Overload**
  - Operator must watch too many screens to determine system state
  - Alarm sensitivity set too high; many false alarms cause operator to ignore alarm altogether



- **Confidence in feedback from HCI**
  - HCI must provide appropriate confidence level to information it is supplying from the system
  - Operator should not be led to trust the monitors too much; they can fail too
  - Redundancy: separate monitors should display information from separate information sources
- **Good HCI design critical in embedded systems**
  - Size, power, cost constraints limit complexity of HCI

Usability versus Safety

- **HCI must be relatively simple for human operator**
  - Intuitive controls
  - Understandable output
- **But making the HCI simpler means the user will be performing repetitive actions**
  - Repetition facilitates human mistakes
  - Safety of the system could be compromised
  - Therac-25 user interface was simplified to make it more usable
- **Usability must be sacrificed to an extent for system safety in the HCI**
  - User must perform unusual actions to commit an operation

Inspection Methods

- **Heuristic Evaluation**
  - Analyzing a design for a user interface and judging it by a set of guidelines that will aid the user to complete his/her task
- **Cognitive Walkthrough**
  - The interface is tracked through the series of steps a user must perform to complete a task
  - Questions are asked at each point to determine if the user has enough information to quickly and accurately complete the task
- **These methods can be applied early in the design phase before the interface is implemented**
- **Extremely tedious and costly to perform for marginal benefit**

Empirical Methods

- **A group of sample users interact with a prototype of the user interface**
  - The users are evaluated on how they perform at the task they must complete and detailed information about what the users did is recorded
  - Much information can be gained from actually testing the interface with a sample group of real users
- **Protocol Analysis**
  - Time-intensive empirical method
  - Massive amounts of data collected
  - Marginal information gained about the UI
- **However, the interface must already be designed and built before it can be tested**
  - Changes are more expensive at later stages

Relationship To Other Topic Areas

- **Safety-Critical Systems/Analysis**
  - Safety-critical systems must account for human operators
  - Humans represent the most unpredictable element in the system and therefore the highest danger to safety
- **Exception Handling**
  - Humans are extremely good at producing exceptional inputs to a system
  - Humans are also generally more flexible at recovering from unanticipated occurrences
- **Security**
  - Robust user interface can inhibit malicious users/operators
- **Social and Legal Concerns**
  - Who is ultimately responsible for system failures? The operator? The designer of the HCI? The company who built the system?

Conclusions & Future Work

◆ Conclusions

- Humans are the most unpredictable part of any system and therefore most difficult to model for HCI design
- Humans make more mistakes under stressful conditions, but are better than nothing
- HCI must provide the appropriate feedback without overloading the user/operator with too much information
- Trade off between making HCI relatively easy to use for humans and ensuring that system safety isn't compromised

◆ Future Work

- Developing metrics to measure defects in and usability of user interfaces (MetriStation here at CMU)
- Focusing more on usability and not safety-critical aspects; this issue needs to be resolved

Safe Systems Through Better UI's

- **Gives several concrete examples of how to make systems safer by improving user interfaces**
- **Major points**
  - Validating Input from the user
  - Monitoring the system
  - Configuring alarm rates
- **Contributions**
  - Usability versus Safety tradeoff
  - Encouraging analysis of safety-critical mistakes for future improvements