Maintenance and Dependability

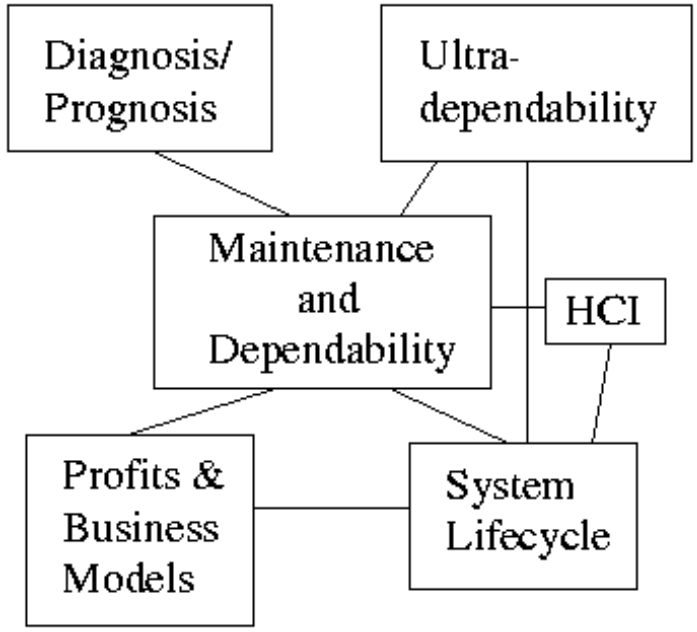
18-849B
March 2, 1999
Adrian Drury

Introduction

- Connections to other topics
- Why do we care?
- Methodology and Techniques
- Conclusions

Relationship To Other Topics

- Diagnosis/prognosis
- Ultradependability
- HCI
- Profits & Business Models
- System Lifecycle



Why Do We Care?

- Consumer model is “cheap and disposable”
- “Dependable embedded systems” are different
- 25% telephone outages caused by telephone company personnel errors
- 47% Navy active duty force and 24% Marine corps assigned to maintenance functions
- Cost - elevator service model vs. car engine controller model?

Relationship to Dependable Embedded Systems

What's likely to cause problems during/after maintenance?

- Operator negligence or error (HCI, Social/Legal concerns)
- Operator stress may be higher than normal (HCI?)
- Upgrade problems
  - Buggy software (Formal methods, software testing)
  - Buggy hardware (Fault tolerance, design verification)

Maintenance Taxonomy

Scheduled vs. On-demand:

- Scheduled - maintenance tasks done regularly (oiling)
- On-demand - done after an indicator changes state (check engine light)

Types of maintenance:

- Preventative - regularly scheduled tasks done whether the system is failing or not.
- Predictive - information gathered about the system is used to set a schedule
- Reactive - maintenance performed after a failure
- Fault-finding - check to see whether the system is still working (check safety systems, for example)

Methodology and Techniques

Evaluation

- Stochastic petri net models
- Markov chain models
- Conjoint simulation
- Bayesian dynamic programming
- Quantitative maintenance analysis

Execution

- Reliability Centered Maintenance (RCM) techniques
  - Very thorough method for ensuring reliability and dependability

Reliability Centered Maintenance

Process

1. Define primary and secondary performance parameters
2. Identify fail states and perform FMEA
3. Identify failure-management policy options
    - Predictive maintenance
    - Preventative maintenance
    - Failure finding
    - Configuration or operation change

Implementation

- Cross-functional review groups
- Software products
- Modeling and scheduling capabilities

Conclusions

- Repair only what you need to, when you need to.
- How do you decide what “need to” is?
- Simulations only get you so far
- Follow a method, such as RCM
- Cost is a big factor - what do you design for?
  Maintenance, no maintenance, easy maintenance
- Designers should help set maintenance schedule