ECE450 – Software Engineering II

Today:

Risk

Risk

- What is risk?
  - "The possibility of suffering loss"
  - Not inherently bad
    - Essential to progress!
  - The challenge is to manage the amount of risk

If you don't actively attack the risks...

...the risks will actively attack you *–Tom Gilb*

Risk Management

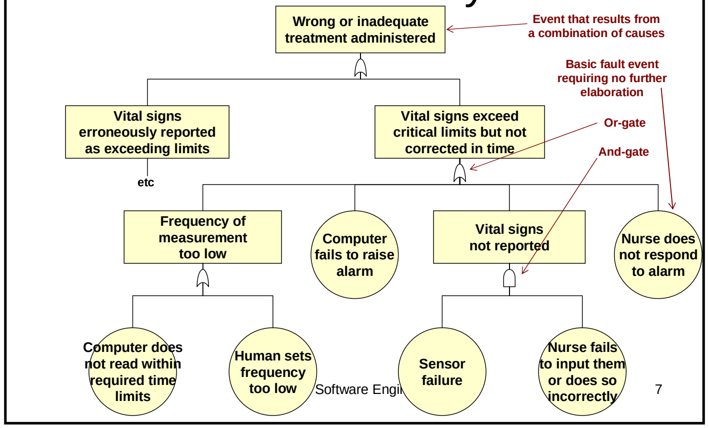
- General idea:
  - Identify your project’s risks
  - Assess their impact and likelihood
  - Devise plans to mitigate or avert them
  - Monitor the risks and their corresponding plans

Risk Identification: Checklists (Boehm’s Top 10 Risks)

- Personnel shortfalls
  - Staff with top talent
  - Job matching
  - Team building
  - Key personnel agreements
  - Cross training
- Unrealistic schedules and budgets
  - Multisource estimation
  - Incremental development
- Developing the wrong features
  - Requirements analysis
  - Prototyping
- Developing the wrong user interface
  - Prototyping, user participation
- Gold-plating
  - Cost-benefit analysis
  - Designing to cost
- Continuing stream of requirements changes
  - High change threshold
  - Deferring changes to later increments
- Shortfalls in external components
  - Benchmarks
  - Inspections
- Shortfalls in external tasks
  - Pre-award audits
  - Award-fee contracts
- Real-time performance shortfalls
  - Simulations
  - Prototyping
- Straining computer science capabilities
  - Technical analysis
  - Cost-benefit analysis
  - Prototyping

Risk Identification: Fault Tree Analysis

adapted from material by Steve Easterbrook. Original work by Leveson (“Safeware”)


Wrong or inadequate treatment administered
Event that results from a combination of causes
Basic fault event requiring no further elaboration
Vital signs erroneously reported as exceeding limits
etc
Vital signs exceed critical limits but not corrected in time
Or-gate
And-gate
Frequency of measurement too low
Computer fails to raise alarm
Vital signs not reported
Nurse does not respond to alarm
Computer does not read within required time limits
Human sets frequency too low
Sensor failure
Nurse fails to input them or does so incorrectly

Risk Assessment

- Quantitative approach:
  - For each risk, **Risk Exposure**:
    - Risk Exposure (RE) = *p*(risk occurring) x loss
- Qualitative approach:
  - Risk exposure matrix *(example from NASA)*:

		Likelihood of Occurrence		
		Very likely	Possible	Unlikely
Undesirable outcome	(5) Loss of Life	Catastro phic	Catastro phic	Severe
	(4) Loss of Spacecra ft	Catastro phic	Severe	Severe
	(3) Loss of Missi on	Severe	Severe	High
	(2) Degrade d Mission	High	Moderate	Low
	(1) Inconv enience	Moderate	Low	Low

Risk Reduction and Aversion

- Quantitative approach:
  - For each mitigation action, **Risk Reduction Leverage**:
    - RRL = (RE_{before} – RE_{after}) / Cost of intervention
- Qualitative approach:
  - Determine Risk Aversion options and a Risk Monitoring Plan:

Risk	Risk Aversion Options	Risk Monitoring
1. Requirements for core module are incorrect	Option 1: Schedule another series of meetings with stakeholders to discuss requirements Option 2: Produce a lo-fi prototype for quick user testing	Track requirements changes on core module, redesign rest of modules immediately
2. No data mining experience in our team	Option 1: Hire data mining professional Option 2: Train Bob (implies losing him for two months)	Track selected strategy after Monday meeting
...	...	...

Risk Monitoring–Top 10 Risks

Risk Item	Weekly Ranking			Risk Resolution Progress
	This	Last	#wks	
Replacing Sensor-Control Software Developer	1	4	2	Top Replacement Candidate Unavailable
Target Hardware Delivery Delays	2	5	2	Procurement Procedural Delays
Sensor Data Formats Undefined	3	3	3	Action Items to Software, Sensor Teams; Due Next Month
Staffing of Design V&V Team	4	2	3	Key Reviewers Committed; Need Fault-Tolerance Reviewer
Software Fault-Tolerance May Compromise Performance	5	1	3	Fault Tolerance Prototype Successful
Accommodate Changes in Data Bus Design	6	-	1	Meeting Scheduled With Data Bus Designers
Testbed Interface Definitions	7	8	3	Some Delays in Action Items; Review Meeting Scheduled
User Interface Uncertainties	8	6	3	User Interface Prototype Successful
TBDs In Experiment Operational Concept	9	7	3	TBDs Resolved
Uncertainties In Reusable Monitoring Software	10	9	3	Required Design Changes Small, Successfully Made

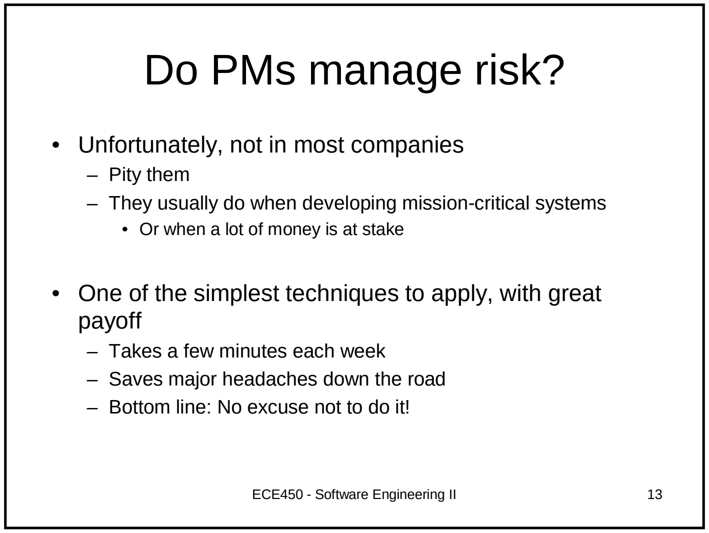
Principles of Risk Management

adapted from material by Steve Easterbrook. Original work by the SEI Continuous Risk Management Guidebook

- Global Perspective
  - View software in context of a larger system
- Forward Looking View
  - Anticipate possible outcomes
  - Identify uncertainty
  - Manage resources accordingly
- Open Communications
  - Free-flowing information at all project levels
  - Value the individual voice
    - Unique knowledge and insights
- Integrated Management
  - Project management is risk management!
- Continuous Process
  - Continually identify and manage risks
- Shared Product Vision
  - Everybody understands the mission (shared ownership)
- Teamwork
  - Work cooperatively to achieve the common goal
  - Pool talent, skills and knowledge

Symptoms of Failure to Manage Risk

- Are overconfidence and complacency common?
  - the Titanic effect - “it can’t happen to us!”
  - Do managers assume it’s safe unless someone proves otherwise?
- Are warning signs routinely ignored?
  - What happens to diagnostic data during operations?
  - Does the organization regularly collect data on anomalies?
  - Are *all* anomalies routinely investigated?
- Is there an assumption that risk decreases?
  - Are successful projects used as an argument to cut safety margins?
- Is there a culture of silence?
  - What is the experience of whistleblowers? (Can you even find any?)



Do PMs manage risk?

- Unfortunately, not in most companies
  - Pity them
  - They usually do when developing mission-critical systems
    - Or when a lot of money is at stake

- One of the simplest techniques to apply, with great payoff
  - Takes a few minutes each week
  - Saves major headaches down the road
  - Bottom line: No excuse not to do it!