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
  – Multi-source 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
• Straining computer science capabilities
  – Technical analysis
  – Cost-benefit analysis
  – Prototyping

Risk Identification:
Fault Tree Analysis

Wrong or inadequate treatment administered

Vital signs erroneously reported as exceeding limits

Vital signs exceed critical limits but not corrected in time

Frequency of measurement too low

Computer fails to raise alarm

Vital signs not reported

Nurse fails to input them or does so incorrectly

ECE450 - Software Engineering II

Risk Assessment

• Quantitative approach:
  – For each risk, Risk Exposure = (risk occurring) x loss

• Qualitative approach:
  – Risk exposure matrix (example from NASA):
Risk Reduction and Aversion

- Quantitative approach:
  - For each mitigation action, **Risk Reduction Leverage (RRL)**:
    - \[ RRL = \frac{RE_{before} - RE_{after}}{Cost\ of\ intervention} \]

- Qualitative approach:
  - Determine Risk Aversion options and a Risk Monitoring Plan:

<table>
<thead>
<tr>
<th>Risk</th>
<th>Risk Aversion Options</th>
<th>Risk Monitoring</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Requirements for core module are incorrect</td>
<td>Option 1: Schedule another series of meetings with stakeholders to discuss requirements</td>
<td>Track requirements changes on core module, redesign rest of modules immediately</td>
</tr>
<tr>
<td>2. No data mining experience in our team</td>
<td>Option 1: Hire data mining professional</td>
<td>Track selected strategy after Monday meeting</td>
</tr>
<tr>
<td></td>
<td>Option 2: Train Bob (implies losing him for two months)</td>
<td></td>
</tr>
</tbody>
</table>

Risk Monitoring—Top 10 Risks

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

Principles of Risk Management

- 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!