Software Project Management
CS6704: Class 8

Instructor: Shawn A. Bohner
Voice: (703) 538-8374
Email: sbohner@vt.edu

Teaching Assistant: Yunxian Zhou
Email: yxzhou@nvc.cs.vt.edu
Voice: (703) 538-8381

Agenda

◆ Discussions
  ● Want to get back to the Midterm...
  ● Midterm Questions/Clarifications
  ● May be a Camera Man Around sometime...

◆ Risk Management: Managing the Right Thing!
  ● Break

◆ Risk Management Practice Improvement
◆ Homework/Project Assignment
Software Risk Management

- The purpose of this module is to define the process of risk management and explain why it is an important part of the planning process for any software project
  - Managing the right thing!
- Objectives
  - Outline software risk management principles
  - Outline key aspects of software project risk
  - Examine how to software risks influence planning
Software Projects are about Risk

- Risk translates to:
  - Lack of information
  - Lack of time, and/or
  - Lack of control
- Risk = F(impact X exposure)
- Failure to act on opportunity can create more future risk

Banker’s Risk  Investor’s Risk  Gambler’s Risk

Basic IT Value Proposition

\[ \text{Value} = \frac{\text{Performance}}{\text{Price}} \]

\textit{Performance (IT Drivers)}
- Reduce Risk
- Increase Customer Satisfaction
- Reduce Cycle Time
- Increase Productivity

\textit{Price (Market)}

\textit{IT value is proportional to the degree it risks or enhances value of the information that it delivers}
Business Perception of IT Value

**Perceived Capability of IT**

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>High</td>
<td>High</td>
<td>Low</td>
<td>Low</td>
</tr>
<tr>
<td>Low</td>
<td>Low</td>
<td>High</td>
<td>High</td>
</tr>
</tbody>
</table>

**Recognized Dependency on Information**

*IT value is both real and perceived – evidence of this perception is in the management measures*

---

**Example: Delivery Risks**

- **Good-Fast-Cheap**: *Pick Good and Fast!*
- **Delivery Risk** is a function of
  - Likelihood of Loss (LL) and Magnitude of Loss (ML)

**Example: Operational Dependability**

\[
Risk = \sum (LL \times ML)
\]

- Many defects: high LL
- Critical defects: high ML
- Few defects: low LL
- Minor defects: low ML

*source*
Example: Delivery Risks (continued)

Now Overlay “Fast” -- Aggressive Schedule associated with Delivery in Internet Time

![Graph showing delivery risks]

- Many defects: high LL
- Critical defects: high ML
- Few defects: low LL
- Minor defects: low ML
- Few rivals: low LL
- Weak rivals: low ML
- Many rivals: high LL
- Strong rivals: high ML

Initiative Risk Likelihood

- Poor Leadership
- Poor Sponsorship
- Technology Mis-Selection
- Inadequate Transition Planning
- Key Staff Turnover
- Poor Coordination with Business
- Cost/Schedule Overruns
- Inadequate Project Planning
- Duplication of effort
- Poor Requirements Management
- Poor Relationship Management
Managing the IT asset portfolio entails examining how much risk an organization can afford and applying the appropriate investment strategy.

Targeting Risk/Return Zone

**Subjective Measures**
- "Customer Satisfaction"
- "Strategic Alignment"
- "Flexibility"
- "Information Quality"

**Objective Measures**
- Productivity in Activity “A”
- Errors in Decision “D”
- Cycle Time of Process “C”
- Project slip rate

**Project Risk/Return**

Stay within the Risk Zone!
Value at Risk: The Architecture

Architecture Bridges the Gap

- Business Architecture
- Information Architecture
- Organization Architecture
- Application Portfolio
- Technology Architecture

Business Strategy
- Environmental Forces
- Business Goals
- Business Policy
- Resource Allocation

Implementation
- Business Processes
- Application Systems
- Technical Infrastructure
- Organizational Structure

Enterprise architecture bridges strategy to implementation and reduces risks.
Business Architecture

- **Business Architecture (BA)** represents an enterprise’s key business strategies and their impact on business functions and processes:
  - Consists of current and future state models of business functions, processes, and information value chains
  - Implemented through the Information and Technical Architectures, and the Applications Portfolio

- A BA must define the “business design” for sustainable evolution and change at the rate of business
Technology Architecture

- **Technology Architecture (TA)** represents an enterprise’s key technical strategies and their impact on business applications and infrastructure:
  - It consists of models of the business domain, information, applications, and technical infrastructure
  - Is implemented through the Information and Technical Infrastructure Architectures, and the Applications Portfolio
  - A TA must define the “technological design” for sustainable competitive advantage and must change at the rate of business

---

Reactive Risk Management

- Project team reacts to risks when they occur
- Mitigation—plan for additional resources in anticipation of fire fighting
- Fix on failure—resource are found and applied when the risk strikes
- Crisis management—failure does not respond to applied resources and project is in jeopardy
Proactive Risk Management

- Formal risk analysis is performed
- Organization corrects the root causes of risk
  - TQM concepts and statistical SQA
  - Examining risk sources that lie beyond the bounds of the software
  - Developing the skill to manage change

Basic Risk Management Paradigm

Diagram showing the cycle of RISK: Identify, Plan, Analyze, Track, Control.
Building a Risk Table

<table>
<thead>
<tr>
<th>Risk</th>
<th>Probability</th>
<th>Impact</th>
<th>RMMM</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Risk Mitigation Monitoring & Management

Building the Risk Table

- Estimate the probability of occurrence
- Estimate the impact on the project on a scale of 1 to 5, where
  - 1 = low impact on project success
  - 5 = catastrophic impact on project success
- sort the table by probability and impact
Risk Mitigation, Monitoring, and Management (RMMM)

- Mitigation—how can we avoid the risk?
- Monitoring—what factors can we track that will enable us to determine if the risk is becoming more or less likely?
- Management—what contingency plans do we have if the risk becomes a reality?

Risk Due to Product Size

Attributes that affect risk:

- Estimated size of the product in LOC or FP?
- Estimated size of product in number of programs, files, transactions?
- Percentage deviation in size of product from average for previous products?
- Size of database created or used by the product?
- Number of users of the product?
- Number of projected changes to the requirements for the product? before delivery? after delivery?
- Amount of reused software?
Risk Due to Business Impact

Attributes that affect risk:

- Affect of this product on company revenue?
- Visibility of this product by senior management?
- Reasonableness of delivery deadline?
- Number of customers who will use this product
- Interoperability constraints
- Sophistication of end users?
- Amount and quality of product documentation that must be produced and delivered to the customer?
- Governmental constraints
- Costs associated with late delivery?
- Costs associated with a defective product?

Risks Due to the Customer

Questions that must be answered:

- Have you worked with the customer in the past?
- Does the customer have a solid idea of requirements?
- Has the customer agreed to spend time with you?
- Is the customer willing to participate in reviews?
- Is the customer technically sophisticated?
- Is the customer willing to let your people do their job—that is, will the customer resist looking over your shoulder during technically detailed work?
- Does the customer understand the software engineering process?
Risks Due to Process Maturity

Questions that must be answered:

- Have you established a common process framework?
- Is it followed by project teams?
- Do you have management support for software engineering?
- Do you have a proactive approach to SQA?
- Do you conduct formal technical reviews?
- Are CASE tools used for analysis, design and testing?
- Are the tools integrated with one another?
- Have document formats been established?

Technology Risks

Questions that must be answered:

- Is the technology new to your organization?
- Are new algorithms, I/O technology required?
- Is new or unproven hardware involved?
- Does the application interface with new software?
- Is a specialized user interface required?
- Is the application radically different?
- Are you using new software engineering methods?
- Are you using unconventional software development methods, such as formal methods, AI-based approaches, artificial neural networks?
- Are there significant performance constraints?
- Is there doubt the functionality requested is "do-able?"
Staff/People Risks

Questions that must be answered:

- Are the best people available?
- Does staff have the right skills?
- Are enough people available?
- Are staff committed for entire duration?
- Will some people work part time?
- Do staff have the right expectations?
- Have staff received necessary training?
- Will turnover among staff be low?

Sample Risk Analysis Sheet

<table>
<thead>
<tr>
<th>System</th>
<th>Impact</th>
<th>Exposure</th>
<th>Resources</th>
<th>Requisites</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bus 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bus 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Infr 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Infr 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Adm 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Adm 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

▲ Risk = Impact x Exposure
- Business Perspective
- Management Perspective
- Technical Perspective
Example Software Project
Risk Reduction Activities

- Thorough risk analysis prior to strategy selection
- Realistic schedule
- Adherence to development standards
- Incremental software development
- Early development of difficult software
- Early compilation of interfaces
- Visibility into timing and sizing from onset
- Extensive use of software progress metrics
- Frequent face-to-face communications
- Periodic audits

Barry Boehm’s Top Software Risks

1. Personnel Shortfalls
2. Unrealistic Schedules and Budgets
3. Developing the wrong software functions
4. Developing the wrong user interface
5. Gold-plating. Requirements scrubbing
6. Continuing stream of requirements changes
7. Shortfalls in externally-performed tasks
8. Shortfalls in externally-furnished components
10. Straining computer science capabilities
Some Risky Observations

- Some formal Risk Management approaches exist, but most are still ad hoc
- Cultural Pattern: Many risks are known to the community developing the system, but work culture inhibits identification, communication, and timely mitigation of program risks
- Same risks appear repeatedly on large organization programs
  - Left unmanaged, they grow up to be real problems
- Difficult to separate software risks from system or program risks
- Good managers are managing risks

Schedule Overrun Risk Areas

- Cost overruns routinely tied to Schedule overruns
- Primary contributors to schedule overruns:
  - Contractual
  - Requirements
  - Management
  - Resources
  - Process
  - Staffing
  - Testing and Integration
## Software Contracts Risks

<table>
<thead>
<tr>
<th>Customer</th>
<th>Provider</th>
</tr>
</thead>
</table>
| - Contract type inconsistent with program risks  
  - Inadequate Cost Estimates  
  - Major milestones imposed without or little vendor input | - Development schedules unrealistic  
  - Contractor milestones inconsistent with software development approach |

### Improvements
- Independent Cost Estimates
- Incorporation of all risks in determining proper contract vehicle and acquisition strategy  
  - Entrance/exit criteria for milestone reviews  
  - Vendor required to identify milestone dates that map to proposed development approach

## Software Requirements Risks

<table>
<thead>
<tr>
<th>Customer</th>
<th>Provider</th>
</tr>
</thead>
</table>
| - Incomplete, ambiguous, unstable system requirements  
  - Untestable system requirements  
  - Inadequate user involvement | - Software requirements incomplete or ambiguous |

### Improvements
- User sign off on requirements  
  - Technical interchange meetings with vendor  
  - Independent quality checks of specifications  
  - Review vendor’s systems engineering organization for software involvement  
  - Entrance/exit criteria for requirements reviews
### Software Management Risks

<table>
<thead>
<tr>
<th>Customer</th>
<th>Provider</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Inadequate visibility into real program status &amp; risks</td>
<td>• Inadequate program plan</td>
</tr>
<tr>
<td>• Inadequate enforcement of development control points</td>
<td>• Inadequate visibility into actual program status</td>
</tr>
<tr>
<td>• Vendor relationship</td>
<td>• Failure to use past trends to predict future</td>
</tr>
<tr>
<td>• External interfaces</td>
<td>• Poor mgmt structure for program</td>
</tr>
<tr>
<td></td>
<td>• Poor subcontractor interface</td>
</tr>
<tr>
<td></td>
<td>• Inadequate Risk Mgt Program</td>
</tr>
</tbody>
</table>

**Improvements**

- Consolidate Risk Management Programs & tie to metrics
- Customer needs sufficient management tools in its organization
- Use Entrance/Exit criteria
- Customer needs to plan on “give-and-take” with provider
- Software audits/Red Teams

### Software Staffing Risks

<table>
<thead>
<tr>
<th>Customer</th>
<th>Provider</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Inadequate technical resources to monitor development</td>
<td>• Inability to “ramp up” at start of contract</td>
</tr>
<tr>
<td>• Inadequate training</td>
<td>• Inadequate staff w/ experience/skills from similar programs</td>
</tr>
<tr>
<td>• Excessive staff turnover</td>
<td>• Staff not familiar</td>
</tr>
</tbody>
</table>

**Improvements**

- Conduct skill assessment to determine readiness to support acquisition; take action to fill gaps
- Training
- Capability Assessment
- Use of Software Engineering Exercise
- Collect metrics on contractor staff turnover
**Software Process Risks**

<table>
<thead>
<tr>
<th>Customer</th>
<th>Provider</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Program Offices often do not have adequate process for acquiring systems</td>
<td>• SEI CMM findings show that most contractors (~50%) are at maturity Level 1</td>
</tr>
</tbody>
</table>

**Improvements**
- Independent audit of customer program office
- Perform assessment of software process (i.e., SEI-CMM or ISO9000) and develop risk reduction program around findings

**Software Testing & Integration Risks**

<table>
<thead>
<tr>
<th>Customer</th>
<th>Provider</th>
</tr>
</thead>
</table>
| • Contract does not require system load test  
• Inadequate fidelity/performance standards for test drivers | • Testing schedule too short & fails to recognize time needed for fixing problems  
• Poor criteria for software test phase transition  
• Inability to use subcontractor skills in system integration |

**Improvement**
- Customer include system load test/scenarios in SOW  
- Vendor criteria for passing from one test phase to another  
- Vendor generates metrics; customer audits metrics process  
- Audit vendor’s schedules for sufficient time to fix errors  
- Customer specifies fidelity
### Resource Software Risks

<table>
<thead>
<tr>
<th>Customer</th>
<th>Provider</th>
</tr>
</thead>
</table>
| • Not enough hardware to maintain adequate maintenance/test facility | • Not enough workstations for staff  
• Not enough hardware to support simultaneous development & testing |

**Improvements**

• Realistic test and fielding plan  
• Red Team to compare test equipment versus test plan

### Homework Assignment for 10/22/01

- **Readings**
  - Read text Chapter 7 on Risk Management
  - Read Paper entitled “Large-Scale Project Management is Risk Management” by Robert Charette
  - Read “Managing Risk in SW Maintenance” by Robert Charette
  - Read “Implementing Risk Management …” by Edmund Conrow

- **Develop a Risk Management Plan for the “Project Management Metrics Database”**

- **Have a great week!**