Risks

Risk analysis and management
Change happens.
Unforeseen circumstances arise.
How should one plan for uncertainty?
True story

A friend of mine was head engineer at a robotics firm CEO negotiated a deal with the US postal service for cleaning robots
Head engineer arrived on site with a working product to discover a major miscommunication

CEO asked him to create corridor-cleaning robots.
USPS asked CEO to create area-cleaning robots.
A week of all-nighters later, they got them!
Overall project risks

- Is management committed?
- Are end-users enthusiastic?
- Are requirements understood?
- Are customers involved and informed?
- Do end-users have realistic expectations?
- Is project scope stable?
- Does the team have the proper skills?
- Are requirements stable?
The four basic risk categories

- Performance risk: will product be fit for intended use?
- Cost risk: will product be created within budget?
- Support risk: will product be easy to adapt?
- Schedule risk: will product be delivered on time?
Risk projection

Establish a scale that indicates the likelihood of risks.
Delineate the consequences of a risk.
Estimate the impact of the risk on the project and product.
Assess the overall accuracy of the risk projection.
Impact assessment

Kinds of impact
- Catastrophic: results in failure of mission.
- Critical: makes primary mission less likely.
- Marginal: might compromise secondary missions.
- Negligible: does not compromise missions.

Kinds of risk
- Performance
- Support
- Cost
- Schedule
Catastrophic risks

Performance: failure to meet requirements leads to mission failure.
Support: failure leads to unsupportable software.
Cost and schedule: failure leads to $500k loss or more in time or revenue or both.
Critical risks

Performance: failure means mission success is questionable
Support: failure means there are minor delays in modification
Cost: failure leads to shortage of financial resources
Schedule: failure leads to increased time-to-market
Marginal risks

Performance: failure leads to minor degradation of secondary missions.
Support: failure leads to responsive software support
Cost: failure is covered by existing financial resources.
Schedule: failure still leaves a realistic, achievable schedule
Negligible risks

Performance: failure does not lead to reductions in technical performance.
Support: failure is easily supportable.
Cost: failure leads to potential budget **underruns**.
Schedule: failure leads to **early** completion.
## The risk matrix: summary

<table>
<thead>
<tr>
<th>Severity\Kind</th>
<th>Performance</th>
<th>Support</th>
<th>Cost</th>
<th>Schedule</th>
</tr>
</thead>
<tbody>
<tr>
<td>Catastrophic</td>
<td>Mission failure</td>
<td>Not supportable</td>
<td>$500K overrun</td>
<td>Unachievable schedule</td>
</tr>
<tr>
<td>Critical</td>
<td>Mission at risk</td>
<td>Delays in modification</td>
<td>$100K-$500K overrun</td>
<td>Possible slippage</td>
</tr>
<tr>
<td>Marginal</td>
<td>Small reduction</td>
<td>Responsive support</td>
<td>Sufficient resources</td>
<td>Achievable schedule</td>
</tr>
<tr>
<td>Negligible</td>
<td>No reduction</td>
<td>Easily supportable</td>
<td>Possible underrun</td>
<td>Possibly ahead of schedule</td>
</tr>
</tbody>
</table>
Risk exposure

Risk exposure
A business name for the **expected value** of the cost of a risk = \( \sum (\text{Prob(failure)} \times \text{Cost(failure)}) \)
(assuming that failures are independent!)
Some politely ignored risks

Some common politely ignored risks
  Employee turnover: will team members remain with the project until completion?
  Employee criticality: will the project fail if an employee leaves?
Job security

Some engineers strive for the wrong kind of job security: being critical to mission. One expert manager I know: "Show me someone who is absolutely critical to operations, and that is the first person I will fire." Presence of critical staff is a catastrophic risk. Your real job as a professional is to avoid your own criticality and program so you can leave the code behind.
The SE and risk

The Software Engineer and risk

**Document** and plan for risks

Risk mitigation, monitoring, and management (RMMM)

Limit **risk exposure**

Understand the dark side of having critical staff
Risk is relative

Suppose your project can take twice as long as expected. Is this a catastrophic risk?
   Yes if schedule is critical to marketing.
   No if not.

Suppose your project can be difficult to support. Is this catastrophic?
   Yes if long product lifecycle.
   No if not.
Some (relatively obvious) risk assessment formulae

Cost of delayed schedule = work lost + revenue lost
Work lost = salary of team * delay time
Revenue lost = estimated sales drop due to delay
(potentially catastrophic if market is lost)

Cost of support deficiency = work lost + revenue lost
Work lost = extra time supporting product * salary of supporters
Revenue lost = estimated sales drop due to support delays (potentially catastrophic if product gets bad press)
Back-of-the-envelope

A back-of-the-envelope calculation on Windows 7
Vista: most unpopular operating system in history
Chrome: threatens Vista with a "friendlier"
alternative.
Captive market: statistically, people are likely to stay
with Windoze even though they hate Vista.
Main risk: losing the market to Chrome.
Accordingly:
To market as fast as possible, whether it works or not!
Estimating probabilities

Estimating project shortfalls: utilize prior history (of company).