Lecture 15:
Scheduling, Estimation, and Prioritization (Part II)

"Once you have a delivery date and a product specification, the main problem is how to control the expenditure of human and technical resources for an on-time delivery of the product."
-- Steve McConnell, *Code Complete*

"Plans are worthless, but planning is everything."
-- Dwight Eisenhower

"Good judgement comes from experience. Experience comes from bad judgement."

31 Jul 2006   CSE403, Summer'06, Lecture 15b   Valentin Razmov

---

Outline

n Software project estimation
n Prioritization
n Scheduling
  n Being behind schedule, ahead of schedule
n Frequent scheduling and prioritization-related mistakes students make
n Best practices for project scheduling
n Scheduling in the context of your projects

31 Jul 2006   CSE403, Summer'06, Lecture 15b   Valentin Razmov

---

Resources

n *The Mythical Man-Month*, by Fred Brooks
  n Ch. 2 (handout)
n *Rapid Development*, by Steve McConnell
  n Ch. 8, 9
  n Ch. 29, 32 (short summaries of best practices)
  n (optional) Ch. 14
n *Death March (2nd ed.)*, by Ed Yourdon
  n Critical-Chain Scheduling (pp.175-177)
n *Code Complete*, by Steve McConnell
n *Software Requirements*, by Karl Wiegers

31 Jul 2006   CSE403, Summer'06, Lecture 15b   Valentin Razmov

---

Scheduling Woes

n If your project moves forward both on budget and on schedule, you are in the minority…
n What can you do if that's not the case?

31 Jul 2006   CSE403, Summer'06, Lecture 15b   Valentin Razmov

---

Your Options If You Fall Behind Schedule

Student Submission

31 Jul 2006   CSE403, Summer'06, Lecture 15b   Valentin Razmov

---

Your Options If You Fall Behind Schedule

Student Submission

Which of these would you choose?
n Negotiate an increase in the amount of time
n Negotiate a reduction in the scope of the project
n Add more people to the team
n Be upfront about it
n Hide it "under the rug" and move forward
n Hope that you can catch up later
n Reduce the outside distractions by moving the team off-site

31 Jul 2006   CSE403, Summer'06, Lecture 15b   Valentin Razmov

Your Options If You Fall Behind Schedule (1/3)

- **Negotiate an increase in the amount of time**
  - Sometimes not an option, but it may be possible
    - Increase by how much? By the slipped time or more?
  - Avoids addressing the bigger problem that caused the delay
- **Negotiate a reduction in the scope of the project**
  - Prioritize the optional and nice-to-have features, then drop the least essential ones.
    - "Which part of 'No!' do you not understand?"
  - Can you save time by providing similar (but perhaps rough) useful functionality?

Your Options If You Fall Behind Schedule (2/3)

- **Add more people to the team**
  - "Adding people to a late software project makes it later." (see Brooks in "Mythical Man Month", 1975)
    - Why? Under what circumstances?
  - There is a limit to schedule compression.

- **Be upfront about it;
  Don't try to "hide it under the rug"**
  - If you conceal the truth, you will almost certainly lose the customer's trust in your team and/or company.

Your Options If You Fall Behind Schedule (3/3)

- **Hope that you can catch up later**
  - Statistics show that this is an illusion if you're more than 10% behind on the schedule.
  - Indeed, you're more likely to fall further behind.

- **Reduce the outside distractions by moving the team off-site**
  - The "skunkworks" team approach
  - May work, if the team is at a stage where it can proceed effectively with very limited communication to the other stakeholders

Your Options If You Are Ahead of Schedule

Student Submission

Your Options If You Are Ahead of Schedule

- **Strive to maintain that edge**
  - Allows you to gently exceed expectations again and again
  - Gives you a "cushion" in case difficulties arise in the future
- **Spend the earned capital by giving everyone on the team extra breathing room**
  - May help if the team really needs it, e.g., near the burn-out point.
- **Push to get even further ahead**
  - May needlessly burn out the team
  - If you manage to get that far ahead of schedule, your original schedule was probably too conservative.
- **Listen to what your stakeholders (incl. upper-level management) say**
  - As much as you may not want to do that…

Frequent Mistakes Students in Previous SE Classes Have Made

Scheduling and prioritization-related:
- Not exploring all unknowns (risks) early on to create a realistic schedule
- Not maintaining an up-to-date schedule with *all* remaining tasks and how they map to the resources (time, people) in the team
- Leaving too few resources (people) for a critical task that can't be delayed
- Not leaving enough "safety net" time before major releases in case something unexpected happens
  - It often happens in the most inopportune moments.

Frequent Mistakes Students in Previous SE Classes Have Made

Scheduling and prioritization-related:

- Underestimating the challenges of a new development environment
  - Overly relying on similarities to known environments
- Spending time on "cool" features that are not central to the needs of the users, while delaying the development of promised features
  - A real project is <u>not</u> about what developers enjoy doing; it's about what brings *value* to customers.
  - The hope is that the two are similar; if not, the latter should take precedence.

Best Practices in Project Scheduling

- **Build in a margin of safety into the schedule**
  - Up to 2x is reasonable and even recommendable, but be careful not to make that practice too widely known!
- **Continuously measure progress and re-estimate resources needed**
  - Daily builds are the "pulse" of your project!
- **Use multiple project estimation approaches and study the differences between them**
- **Scrub the requirements**

Scheduling in the Context of Your Team's Project

- According to data in *Code Complete*, the breakup of development time for a 10-15KLOC project is:
  - 13% - architecture
  - 20% - detailed design
  - 20% - coding and debugging
  - 20% - unit testing
  - 12% - integration
  - 15% - system testing
- Is this reflected on your latest schedule?
- How far into each phase is your project?
- Whose job is it to take care of scheduling on your team? Who owns and manages the schedule?

Scheduling in the Context of Your Team's Project (cont.)

- <u>Case</u>: Your team has a fixed delivery date and an existing product specification, as well as relatively fixed (but flexible) human resources. What can you do if the latest project estimate suddenly reveals that you cannot deliver for another 3 weeks?

Scheduling in the Context of Your Team's Project (cont.)

- <u>Case</u>: Your team has a fixed delivery date and an existing product specification, as well as relatively fixed (but flexible) human resources. What can you do if the latest project estimate suddenly reveals that you cannot deliver for another 3 weeks?
- **Renegotiate (reduce) functionality**
  - Push certain non-essential features to version 2
  - Adjustments are non-linear for changes >10% of the original estimates
- **Use outside technical resources**
  - Reuse code, delegate tasks, etc.
- **Restructure work environment**
  - Skunkworks team?
- ?

Favorite Related Quotes

- "Doing things at the last minute is much more expensive than just before the last minute." (Randy Pausch)
- "If you haven't got time to do it right, you don't have time to do it wrong."
- "Good judgement comes from experience. Experience comes from bad judgement."
- "Failing to plan is planning to fail."
- "Work expands so as to fill the time available for its completion." (Parkinson's Law, 1957)

Lecture 16:
Personality Types and
Miscommunication Issues
(Part II)

> "*And so these men of Indostan*
> *Disputed loud and long,*
> *Each in his own opinion*
> *Exceeding stiff and strong,*
> *Though each was partly in the right,*
> *And all were partly in the wrong!"*
> *-- John Godfrey Saxe*

Outline

n MBTI personality test and results
n Stereotyping and discrimination
n Examples of miscommunication
   n … and its consequences

Resources

n "*The Blind Men and the Elephant*", by John Godfrey Saxe (via David Schmaltz)
n http://www.anecdotov.net/

Take-Away Points from Last Time

n There are people who are different from you but still perfectly reasonable.
   n Working with such people requires mutual respect and understanding. Effective communication is a prerequisite for this.

n Stereotyping is counterproductive: it closes doors and eliminates desirable possibilities.
   n "When you label me, you negate me."

The Blind Men and the Elephant

n How does this relate to software engineering?

n How does it relate to the discussion on personality types?

Project (Mis)communication

n 1. What the project manager proposed

1. Как было предложено организатором разработки

Project (Mis)communication

- n 1. What the project manager proposed
- n 2. What the technical specification stated

31 Jul 2006

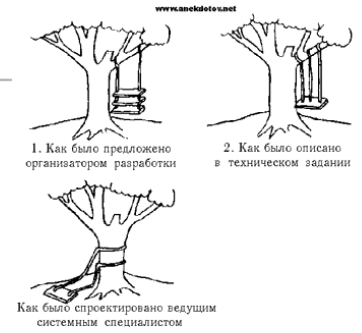## Project (Mis)communication

- n 1. What the project manager proposed
- n 2. What the technical specification stated
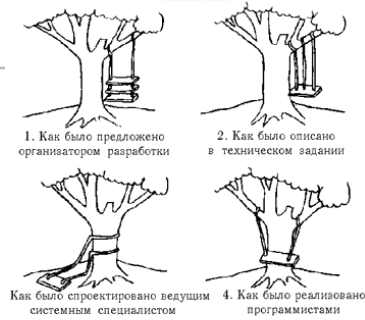- n 3. What the lead system specialist designed

31 Jul 2006

Project (Mis)communication

- n 1. What the project manager proposed
- n 2. What the technical specification stated
- n 3. What the lead system specialist designed
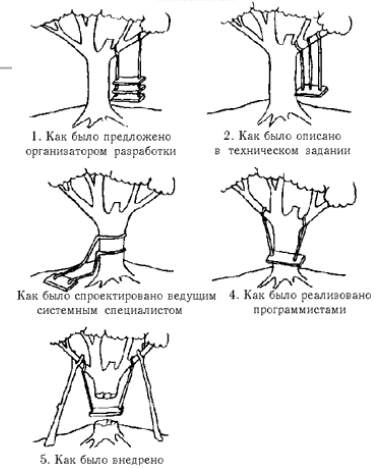- n 4. What the programmers implemented

31 Jul 2006

Project (Mis)communication

- n 1. What the project manager proposed
- n 2. What the technical specification stated
- n 3. What the lead system specialist designed
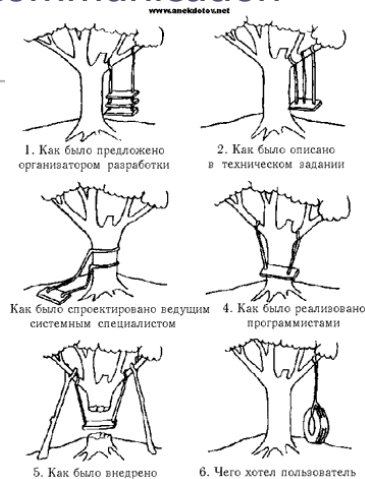- n 4. What the programmers implemented
- n 5. How the product was deployed

31 Jul 2006

Project (Mis)communication

- n 1. What the project manager proposed
- n 2. What the technical specification stated
- n 3. What the lead system specialist designed
- n 4. What the programmers implemented
- n 5. How the product was deployed
- n 6. What the customer had asked for

31 Jul 2006