CSE 403
Lecture 3

Teams and Software

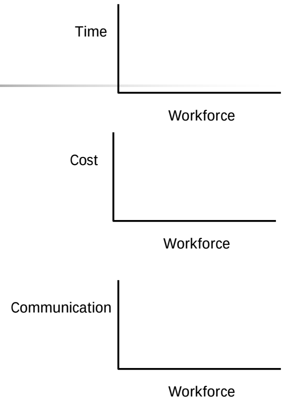
Success Criteria Discussion

Why teams?

- What is a team?

Team size

- Bigger is better
- Smaller is better

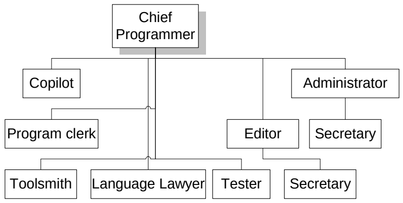


Team structure

- Many different models

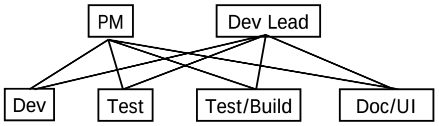
Software development teams

- Brooks
  - Surgeon team



Chief programmer team

- Key points
  - Technical leadership
  - Technical lead has direct development responsibilities



Successful software teams

- Studies show a 10 to 1 difference in productivity of programmers
- Equal differences observed in productivity of teams
- Substantial differences observed in performances of teams *even when the strength of the programmers is equivalent*

What makes a successful team?

- Shared, elevating vision or goal
- Team identity
- Results driven structure
- Competent team members
- Commitment to the team
- Mutual trust
- Interdependence among team members
- Effective communication
- Sense of autonomy
- Sense of empowerment
- Small team size
- High level of enjoyment

Team building

- Team members derive satisfaction from the team's accomplishments
- Important to both
  - Reward the team's success
  - Maintain individual accountability

Motivation

- Motivation is undoubtedly the single greatest influence on how well people perform. Most productivity studies have found that motivation has a stronger influence on productivity than any other factor. (Boehm 1981)

Top five motivation factors

- Achievement
  - Ownership
  - Goal setting
- Possibility for Growth
- Work itself
  - Skill variety
  - Task identity
  - Task significance
  - Autonomy
  - Job feedback

Top five motivation factors

- Opportunity to focus on the work itself
  - Reduce administration
  - Remove obstacles
- Personal life
- Technical-supervision opportunity
  - Assign each developer to be technical lead for some particular product area

How not to manage: Morale killers

- Management manipulation
- Excessive scheduling pressure
- Lack of appreciation for team member's efforts
- Inappropriate involvement of technically inept management
- Not involving team members in decisions that affect them
- Productivity barriers
- Low quality
- Heavy handed motivation campaigns
- Frequent changes in direction

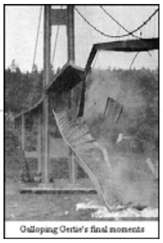
Student Project Teams

- I've observed a high success rate
- But failures happen . . .

Is Software Different?

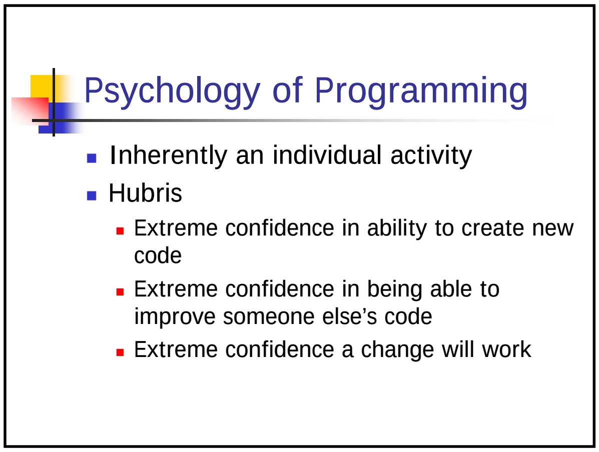
- Software Projects have long had to reputation for failing
- In the 1970's a "Software Crisis" was identified
- Many examples of big software projects that have failed
- Many examples of catastrophic failures caused by software

It's not just software . . .


Galloping Gertie's final moments

Is the engineering of software worse than other fields?

- Less history
- Less professionalism
- More of a craft
- Technology changing rapidly
- Impossible to manage programmers
- Difficulty in estimating costs
- Low cost of failure
- Low cost of change



Psychology of Programming

- Inherently an individual activity
- Hubris
  - Extreme confidence in ability to create new code
  - Extreme confidence in being able to improve someone else’s code
  - Extreme confidence a change will work