The Surgical Team

Productivity Variations

- Measured performance of a group of programmers **experienced**
- Ratios between best and worst performances:
  - 10:1 on productivity measurements
  - 5:1 on program speed and space measurements
- A $20,000/year programmer may well be 10 times as productive as a $10,000/year programmer.
  - The converse may be true, too.

As Few Minds a system

> built by as few Minds as possible

- 200 man + 25 manager
  → fire 175 troops!
- small sharp team (~10 people)
- 그렇지만 여전히 manpower이 필요하다.

  → 소수가 design, 다수가 구현

Mill's Proposal

> Each segment of a large job is tackled by a team
> each team should be organized like a
> **surgical team**

- Surgeon
  - chief
  - define specifications
  - 10y~
- Copilot

    - less experienced
    - “insurance against disaster to the surgeon”
- Toolsmith
    - see to the tools for surgeon
- Tester
- Administrator
- Editor
- ...

Common Pattern

- Sergeon
  + Administrator + Editor
  + Copilot (+toolsmith+tester)
- No Democracy!

Example: Amazon

- 5~8 per team
- design, implementation, build, test, deploy
    - no separate test team
    - → reduce communication overhead

Documentation

> minimize time for communication

- UML이라는 디자인용 notation이 있긴 하지만 잘 쓰지는 않는다 (너무 복잡)

No Silver Bullet

> ***“There is no single development, in either technology or management technique, which by itself promises even one order-of-magnitude improvement within a decade in productivity, in reliability, in simplicity.”***

Werewolf

- Usually innocent and straightforward
  - first design
  - then build
  - finally test :-)
- Capable of becoming a monster of:
  - missed schedules
  - blown budgets
  - flawed products

> **Is there any silver bullet for this type of werewolf?**
> **No**

Difficulties

- Essential: specification, design, testing
  - cannot be eliminated
- accidental: representing the construct
  - use high-level language!
  - unify programming environments!

hope for silver bullet (1n 1986...)

- Ada and other high-level language advances
- Cf. Ada Lovelace, the first computer programmer
- Object-oriented programming
- Artificial intelligence
- Expert systems
- “Automatic” programming
- Graphical programming
- Program verification
- There is no magic here.
- Environments and tools
- Workstations
-