Foundations of Software Engineering

Part 24: Teams
Michael Hilton
administrivia

• HW5 due tonight
• HW6 released today
  – TLDR; make a contribution to an open-source project
Case Studies

Disclaimer: All pictures represent abstract developer groups or products to give a sense of scale; they are not necessarily the developers of those products or developers at all.
How to structure teams?

- Microblogging platform; 3 friends
How to structure teams?

• Banking app; 15 developers
How to structure teams?

- Mobile game;
  50ish developers;
- distributed teams?
How to structure teams?

- Mobile game; 200ish developers
How to structure teams?

- Ride sharing app and self-driving cars; 1200 developers; 4 sites
Teams
Necessity of Groups

• Division of labor
• Division of expertise (e.g., security expert, database expert)
Team Issues

• Social loafing
• Groupthink
• Multiple/conflicting goals
• Process costs
Team issues: Social loafing
Social loafing

• People exerting less effort within a group
• Reasons
  – Diffusion of responsibility
  – Motivation
  – Dispensability of effort / missing recognition
  – Avoid pulling everybody / "sucker effect"
  – Submaximal goal setting
• “Evaluation potential, expectations of co-worker performance, task meaningfulness, and culture had especially strong influence”

Mitigation Strategies

• Involve all team members, co-location
• Assign specific tasks with individual responsibility
  – Increase identifiability
  – Team contracts, measurement
• Provide choices in selecting tasks
• Promote involvement, challenge developers
• Reviews and feedback
• Team cohesion, team forming exercises
• Small teams
Agile Practices as Mitigation?
Responsibilities & Buy-In

• Involve team members in decision making
• Assign responsibilities (ideally goals not tasks)
• Record decisions and commitments; make record available
Team issues: Groupthink
WE NEED MORE DISSenting OPINIONS.

WE AGREE 100%
Groupthink

- Group minimizing conflict
- Avoid exploring alternatives
- Suppressing dissenting views
- Isolating from outside influences
- -> Irrational/dysfunctional decision making
**Star Wars: Episode I - The Phantom Menace** (1999)

🌟 55% 🛡️ 59%

Critics Consensus: Burdened by exposition and populated with stock characters, *The Phantom Menace* gets the Star Wars prequels off to a bumpy – albeit visually dazzling – start.

Starring: Liam Neeson, Ewan McGregor, Natalie Portman

Director: George Lucas

**Star Wars: Episode VI - Return of the Jedi** (1983)

 Decorating 80% 🚀 94%

Critics Consensus: Though failing to reach the cinematic heights of its predecessors, *Return of the Jedi* remains an entertaining sci-fi adventure and a fitting end to the classic trilogy.

Starring: Mark Hamill, Carrie Fisher, Harrison Ford

Director: Richard Marquand

**Star Wars: Episode V - The Empire Strikes Back** (1980)

🌟 95% 🛡️ 97%

Critics Consensus: Dark, sinister, but ultimately even more involving than *A New Hope*, *The Empire Strikes Back* defies viewer expectations and takes the series to heightened emotional levels.

Starring: Mark Hamill, Harrison Ford, Carrie Fisher

Director: Irvin Kershner

**Star Wars: Episode IV - A New Hope** (1977)

🌟 93% 🛡️ 96%

Critics Consensus: A legendarily expansive and ambitious start to the sci-fi saga, George Lucas opened our eyes to the possibilities of blockbuster filmmaking and things have never been the same.

Starring: Mark Hamill, Harrison Ford, Carrie Fisher

Director: George Lucas
Time and Cost Estimation
Causes of Groupthink

• High group cohesiveness, homogeneity
• Structural faults (insulation, biased leadership, lack of methodological exploration)
• Situational context (stressful external threats, recent failures, moral dilemmas)
Symptoms

• Overestimation of ability
  – invulnerability, unquestioned believe in morality

• Closed-mindedness
  – ignore warnings, stereotyping
  – innovation averse

• Pressure toward uniformity
  – self-censorship, illusion of unanimity, …
Studies Show

• Gender-diverse management teams showed superior return on equity, debt/equity ratios, price/equity ratios, and average growth. -Rohner, U. and B. Dougan (2012)

• Gender-balanced teams were the most likely to experiment, be creative, share knowledge, and fulfill tasks. -Lehman Brothers Center for Women in Business. (2008)

• Gender diversity on technical work teams was associated with superior adherence to project schedules, lower project costs, higher employee performance ratings, and higher employee pay bonuses. -Turner, L. (2009)
Unconscious Bias

We all have shortcuts, or “schemas,” that help us make sense of the world. But our shortcuts sometimes make us misinterpret or miss things. That’s unconscious bias.
Unconscious bias

• Pervasive, cultural
• Raise awareness
• Explicit goals
• Measurement
Mitigation Strategies

• Several agile techniques
  – Planning poker
  – Tests, continuous integration
  – On-site customers

• Diverse teams

• Management style

• Avoid HR evaluation by metrics

• Separate QA from development

• Outside experts

• Process reflection

• …
Practical Help

national center for women & information technology
Team issues: Multiple/conflicting goals
How can I Make it?

How can I Break it?
Incentives?

• Team incentives
• vs individual incentives?
Team issues: Process costs
Mythical Man Month

- Brooks's law: *Adding manpower to a late software project makes it later*

1975, describing experience at IBM developing OS/360
Process Costs

\[ n(n - 1) / 2 \]

communication links
Process Costs
Brook's Surgical Teams

- Chief programmer – most programming and initial documentation
- Support staff
  - Copilot: supports chief programmer in development tasks, represents team at meetings
  - Administrator: manages people, hardware and other resources
  - Editor: editing documentation
  - Two secretaries: one each for the administrator and editor
  - Program clerk: keeps records of source code and documentation
  - Toolsmith: builds specialized programming tools
  - Tester: develops and runs tests
  - Language lawyer: expert in programming languages, provides advice on producing optimal code.
Microsoft's Small Team Practices

• Vision statement and milestones (2-4 month), no formal spec
• Feature selection, prioritized by market, assigned to milestones
• Modular architecture
  – Allows small federated teams (Conway's law)
• Small teams of overlapping functional specialists

Windows 95: 200 developers and testers, one of 250 products
Microsoft's Small Team Practices

- Feature Team
  - 3-8 developers (design, develop)
  - 3-8 testers (validation, verification, usability, market analysis)
  - 1 program manager (vision, schedule communication; leader, facilitator) – working on several features
  - 1 product manager (marketing research, plan, betas)
Microsoft's Small Team Practices

• "Synchronize and stabilize"

• For each milestone
  – 6-10 weeks feature development and continuous testing
    • frequent merges, daily builds
  – 2-5 weeks integration and testing ("zero- bug release", external betas )
  – 2-5 weeks buffer
Agile Practices (e.g., Scrum)

• 7+/-2 team members, collocated
• Self managing
• Scrum master (rotating role)
• Product owner / customer representative
Mantle and Lichty

• Ideal team size: 2-3 co-located developers if possible
Large teams (29 people) create around six times as many defects as small teams (3 people) and obviously burn through a lot more money. Yet, the large team appears to produce about the same mount of output in only an average of 12 days’ less time. This is a truly astonishing finding, through it fits with my personal experience on projects over 35 years.

- Phillip Amour, 2006, CACM 49:9
Establish communication patterns

• Avoid overhead
• Ensure reliability
• Constraint latency

• e.g. Issue tracker vs email; online vs face to face
Awareness

• Notifications
• Brook's documentation book
• Email to all
• Code reviews
Conway’s Law

“Any organization that designs a system (defined broadly) will produce a design whose structure is a copy of the organization's communication structure.”

— Mel Conway, 1967

“If you have four groups working on a compiler, you'll get a 4-pass compiler.”
Congruence
Socio-Technical Congruence

• Structural congruence
• Geographical congruence
• Task congruence
• IRC communication congruence
Teamwork Guidelines

• Respect Conway's Law
  – Code structure and team structure should align

• Seek well-defined, stable interfaces
Agile Practices as Mitigation?
Matrix Organization

Temporary assignment to projects; flexible staffing
Project Organization

- mgmt
- System programmers
- Application programmers
- QA
- Security
- Marketing

- Project 1
- Project 2
- Project 3
Case Study: Brøderbund

• As the functional departments grew, staffing the heavily matrixed projects became more and more of a nightmare. To address this, the company reorganized itself into “Studios”, each with dedicated resources for each of the major functional areas reporting up to a Studio manager. Given direct responsibility for performance and compensation, Studio managers could allocate resources freely.

• The Studios were able to exert more direct control on the projects and team members, but not without a cost. The major problem that emerged from Brøderbund’s Studio reorganization was that members of the various functional disciplines began to lose touch with their functional counterparts. Experience wasn’t shared as easily. Over time, duplicate effort began to appear.
Case Study
Commitment & Accountability

• Conflict is useful, expose all views
• Come to decision, commit to it
• Assign responsibilities
• Record decisions and commitments; make record available
Bell & Hart – 8 Causes of Conflict

• Conflicting resources.
• Conflicting styles.
• Conflicting perceptions.
• Conflicting goals.
• Conflicting pressures.
• Conflicting roles.
• Different personal values.
• Unpredictable policies.

https://www.mindtools.com/pages/article/eight-causes-conflict.htm
Virtual Teams
Virtual Teams?
Computer Supported Collaborative Work (CSCW): Technology-assisted collaboration

• Many failures

• Isolated, but very significant, success
  — Jazz, Github, ...
Spotify Squads
Principles

• Rules are a good start, then break them when needed
• Agile > Scrum
• Principles > Practices
• Autonomy, Mastery, Purpose
• *Be autonomous, but don’t sub-optimize!*
Autonomous Squads
Aligned Autonomous squads

- High alignment, low autonomy: We need to cross the river.
- Low alignment, low autonomy: We need to build a bridge.
- High alignment, high autonomy: Figure out how.
- Low alignment, high autonomy: Hope someone is working on the problem...
Squads, Tribes, Chapters, Guilds
Getting into production
Decouple teams and releases
Context

Spotify Boasts 140M Active Users, 50M Premium Subs
Worldwide monthly active users and paid subscribers of Spotify (in millions)

Source: Spotify
Discussion

• Benefits?
• Challenges?
• Implementation pitfalls?
General Guidelines
Hints for team functioning

• Trust them; strategic not tactical direction
• Reduce bureaucracy, protect team
• Physical colocation, time for interaction
• Avoid in-team competition (bonuses etc)
• Time for quality assurance, cult of quality
• Realistic deadlines
• Peer coaching
• Sense of elitism
• Allow and encourage heterogenity

DeMarco and Lister. Peopleware. Chapter 23
Team Fusion

- Forming, Storming, Norming, Performing
- Preserve existing teams, resist project mobility
Elitism Case Study: The Black Team

• Legendary team at IBM in the 1960s
• Group of talented ("slightly better") testers
  – Goal: Final testing of critical software before delivery
• Improvement over first year
• Formed team personality and energy
  – "adversary philosophy of testing"
  – Cultivated image of destroyers
  – Started to dress in black, crackled laughs, grew mustaches
• Team survived loss of original members

DeMarco and Lister. Peopleware. Chapter 22
Troubleshooting Teams

- Cynicism as warning sign
- Training to improve practices
- Getting to know each other; celebrate success; bonding over meals
- “A meeting without notes is a meeting that never happened”
Further Reading

• Mantle and Lichty. Managing the Unmanageable. Addison-Wesley, 2013
  – Very accessible and practical tips at recruiting and management

  – Anecdotes, stories, and tips on facilitating teams, projects, and environments

• Sommerville. Software Engineering. 8th Edition. Chapter 25