Software Project Management
Fifth Edition

Chapter 12

Working in teams

McGraw Hill Education

Becoming a team

Five basic stages of development:

- Forming
- Storming
- Norming
- Peforming
- Adjourning

Classification associated with Tuckman and Jensen

Balanced teams

Meredith Belbin studied the performance of top executives carrying out group work at the Hendon Management Centre

Tried putting the 'best' people together in 'Apollo' teams – almost invariably did badly

Identified the need for a balance of skills and management roles in a successful team

Management team roles

The co-ordinator – good at chairing meetings

The 'plant' – an idea generator

The monitor-evaluator – good at evaluating ideas

The shaper – helps direct team's efforts

The team worker – skilled at creating a good working environment

Belbin management roles - continued

The resource investigator – adept at finding resources, including information

The completer-finisher – concerned with getting tasks completed

The implementer – a good team player who is willing to undertake less attractive tasks if they are needed for team success

The specialist – the 'techie' who likes to acquire knowledge for its own sake

Group performance

Some tasks are better carried out collectively while other tasks are better delegated to individuals

Additive tasks – the effort of each participant is summed

Compensatory tasks – the judgements of individual group members are summed – errors of some compensated for by judgements of others

Group performance - continued

Disjunctive tasks – there is only one correct answer – someone must:

Come up with right answer

Persuade the other that they are right

Conjunctive – the task is only finished when all components have been completed

'Social loafing'

Tendency for some team participants to 'coast' and let others do the work

Also tendency not to assist other team members who have problems

Suggested counter-measures:

   Make individual contributions identifiable

   Consciously involve group members ('loafer' could in fact just be shy!)

   Reward 'team players'

Barriers to good team decisions

Inter-personal conflicts – see earlier section on team formation

Conflicts tend to be a dampened by emergence of *group norms* – shared group opinions and attitudes

Risky shift – people in groups are more likely to make risky decisions than they would as individuals

Delphi approach

To avoid dominant personalities intruding the following approach is adopted

  Enlist co-operation of experts

  Moderator presents experts with problem

  Experts send in their recommendations to the moderator

  Recommendations are collated and circulated to all experts

  Experts comment on ideas of others and modify their own recommendation if so moved

  If moderator detects a consensus, stop; else back to 4

Team 'heedfulness'

Where group members are aware of the activities of other members that contribute to overall group success

Impression of a 'collective mind'

Some attempts to promote this:

- Egoless programming
- Chief programmer teams
- XP
- Scrum

Egoless programming

Gerry Weinberg noted a tendency for programmers to be protective of their code and to resist perceived criticisms by others of the code

Encouraged programmers to read each others code

Argued that software should become communal, not personal – hence 'egoless programming'

Organization and Team Structures

Two important issues that are critical to the effective functioning of every organization are:

Department structure: How is a department organized into teams?

Team structure: How are the individual project teams structured?
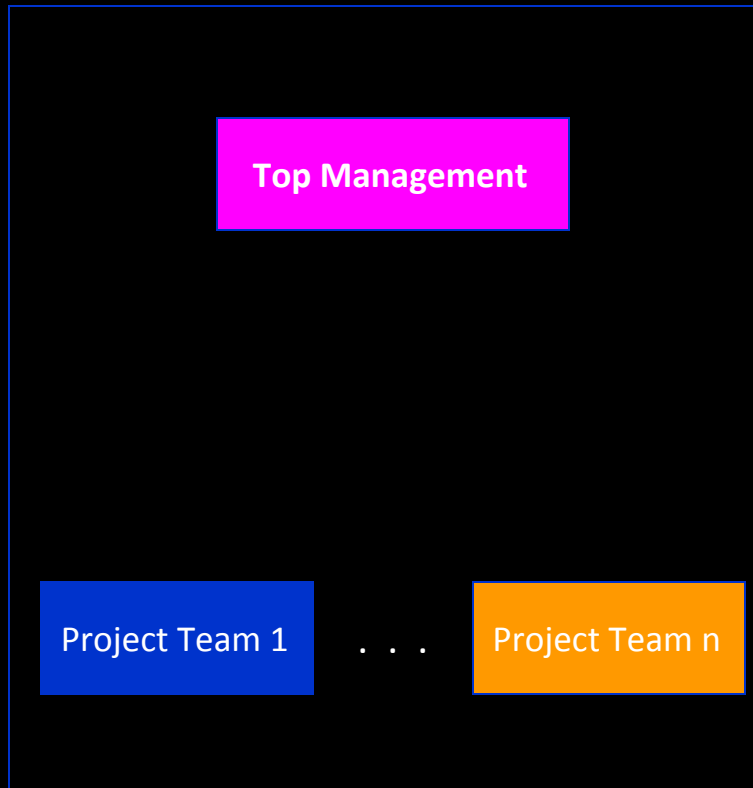
Department Structure

Functional format:

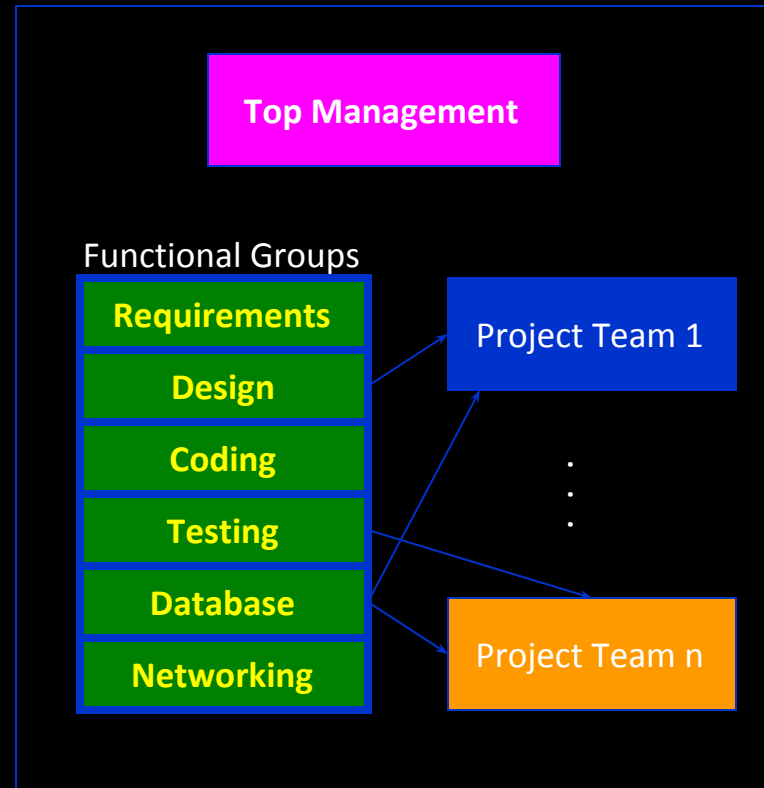Each functional group comprises of developers having expertise in some specific task or functional area.

Project format:

The same team carries out all the project activities.

McGraw Hill Education

Functional and Project Formats

(a) Project Organization

(b) Functional Organization

Functional versus project formats

Ease of staffing

Production of good quality documents

Job specialization

Efficient handling of the problems associated with manpower turnover

Career planning

Matrix Format

- The pool of functional specialists are assigned to different projects as needed.

Functional group	Project			
	#1	#2	#3	
#1	2	0	3	Functional manager 1
#2	0	5	3	Functional manager 2
#3	0	4	2	Functional manager 3
#4	1	4	0	Functional manager 4
#5	0	4	6	Functional manager 5
	Project manager 1	Project manager 2	Project manager 3	

McGraw Hill Education

Team Structure

We consider only three team structures:
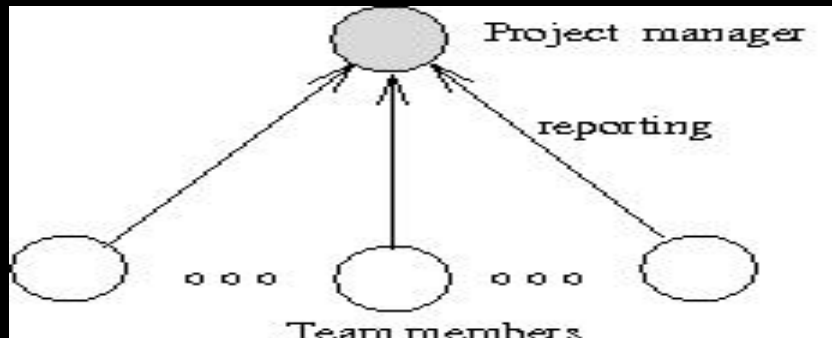
Democratic,

Chief programmer,

Mixed team

Chief programmer teams

Fred Brooks was concerned about the need to maintain 'design consistency' in large software systems

Appointment of key programmers, **Chief Programmers**, with responsibilities for defining requirements, designing, writing and test software code

Assisted by a support team: **co-pilot** – shared coding, **editor** who made typed in new or changed code, **program clerk** who wrote and maintained documentation and **tester**
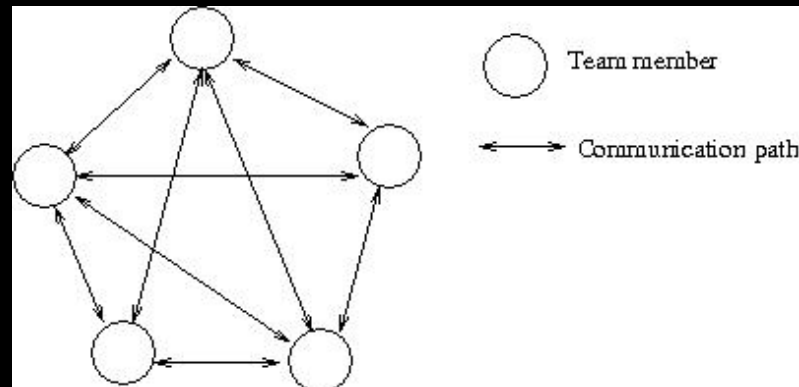
Problem – finding staff capable of the chief programmer role

Democratic Team

Does not enforce any formal team hierarchy.

Decisions are taken based on discussions,

any member is free to discuss with any other member

Since a lot of debate and discussions among the team members takes place,

for large team sizes significant overhead is incurred

Mixed Control Team Structure

Incorporates both hierarchical reporting and democratic set up.
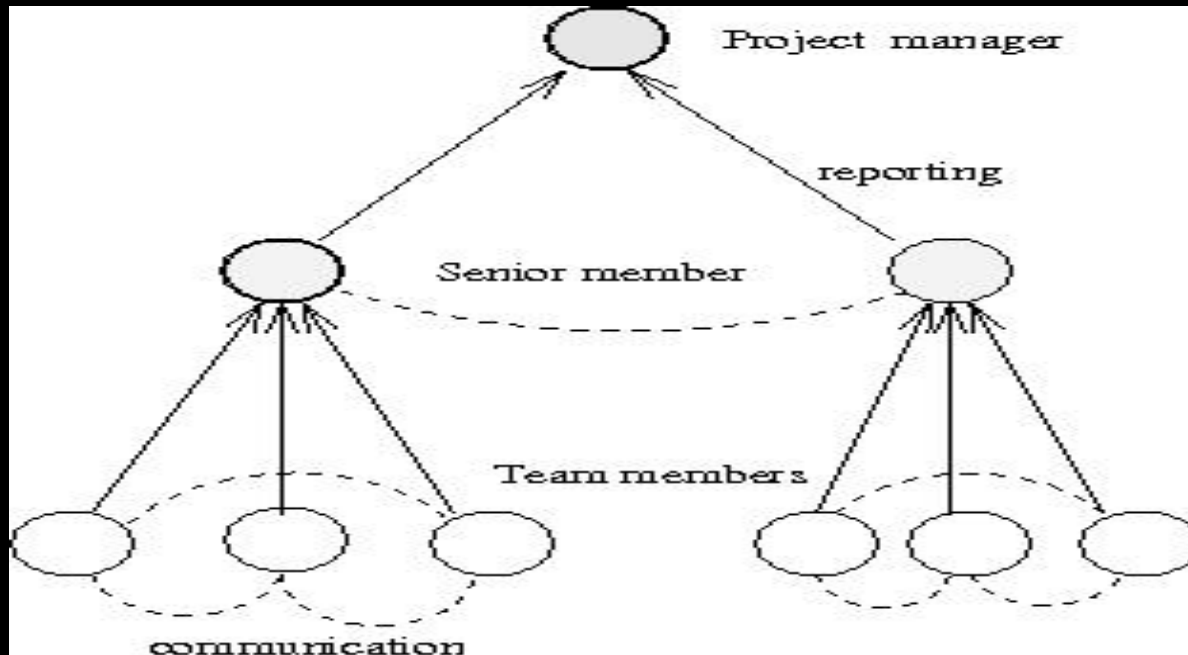
Extreme programming

XP can be seen as an attempt to improve team heedfulness and reduce the length of communication paths (the time between something being recorded and it being used)

Software code enhanced to be self-documenting

Software regularly refactored to clarify its structure

Test cases/expected results created *before* coding – acts as a supplementary specification

Pair programming – a development of the co-pilot concept

Scrum

Named as an analogy to a rugby scrum – all pushing together

Originally designed for new product development where 'time-to-market' is important

'Sprints' increments of typically one to four weeks

Daily 'scrums' – daily stand-up meetings of about 15 minutes

Scrum - continued

Unlike XP, requirements are frozen during a sprint

At the beginning of the sprint there is a sprint planning meeting where requirements are prioritized

At end of sprint, a review meeting where work is reviewed and requirements may be changed or added to

Co-ordination of dependencies

The previous discussion on team heedfulness focused (mainly) in communication inside the team

What sort of communications are needed between teams and other units

Co-ordination theory has identified the following types of coordination:

Shared resources. e.g. where several projects need the services of scarce technical experts for certain parts of the project.

Producer-customer ('right time') relationships. A project activity may depend on a product being delivered first.

Task-subtask dependencies. In order to complete a task a sequence of subtasks have to be carried out.

Coordination of dependencies - continued

Accessibility ('right place') dependencies. This type of dependency is of more relevance to activities that require movement over a large geographical area, but arranging the delivery and installation of IT equipment might be identified as such.

Usability ('right thing') dependencies. Broader concern than the design of user interfaces: relates to the general question of *fitness for purpose,* e.g. the satisfaction of business requirements.

Fit requirements. This is ensuring that different system components work together effectively.

Why 'virtual projects'?

The physical needs of software developers (according to an IBM report):

100 square feet of floor space

30 square feet of work surface

Dividers at least 6 feet high to muffle noise

Demarco and Lister found clear statistical links between noise and coding error rates

One answer: send the developers home!

Possible advantages

Can use staff from developing countries – lower costs

Can use short term contracts:

  Reduction in overheads related to use of premises

  Reduction in staff costs, training, holidays, pensions etc.

Can use specialist staff for specific jobs

Further advantages

Productivity of home workers can be higher – fewer distractions

Can take advantage of time zone differences e.g. overnight system testing

McGraw Hill Education

Some challenges

Work requirements have to be carefully specified

Procedures need to be formally documented

Co-ordination can be difficult

Payment methods need to be modified – piece-rates or fixed price, rather then day-rates

More challenges

Possible lack of trust when there is no face-to-face contact

Assessment of quality of delivered products needs to be rigorous

Different time zones can cause communication and co-ordination problems

Time/place constraints on communication

	Same place	Different place
Same time	Meetings, interviews	Telephone, Instant messaging
Different times	Notice boards Pigeon-holes	Email Voicemail Documents

McGraw Hill **Education**

Other factors influencing communication genres

Size and complexity of information – favours documents

Familiarity of context e.g. terminology – where low, two-way communication favoured

Personally sensitive – it has to be face-to-face communication here

Best method of communication depends on stage of project

Early stages

    Need to build trust

    Establishing context

    Making important 'global' decisions

    Favours same time/ same place

Intermediate stages

    Often involves the paralled detailed design of components

    Need for clarification of interfaces etc

    Favours same time/different place

Best method of communication depends on stage of project

Implementation stages

   Design is relatively clear

   Domain and context familiar

   Small amounts of operational data need to be exchanged

   Favours different time/different place communications e.g. e-mail

Face to face co-ordination meetings – the 'heartbeat' of the project

Communications plans

As we have seen choosing the right communication methods is crucial in a project

Therefore, a good idea to create a **communication plan**

Stages of creating a communication plan

    Identify all the major stakeholders for the project – see chapter 1

    Create a plan for the project – see chapter 3

    Identify stakeholder and communication needs for each stage of the project

    Document in a communication plan

Content of a communication plan

For each communication event and channel, identify:

What. This contains the name of a particular communication event, e.g, 'kick-off meeting', or channel, e.g. 'project intranet site'.

Who/target. The target audience for the communication.

Purpose. What the communication is to achieve.

When/frequency. If the communication is by means of a single event, then a date can be supplied. If the event is a recurring one, such as a progress meeting then the frequency should be indicated.

Type/method. The nature of the communication, e.g., a meeting or a distributed document.

Responsibility. The person who initiates the communication.

Leadership: types of authority

Position power

    Coercive power – able to threaten punishment

    Connection power – have access to those who do have power

    Legitimate power – based on a person's title conferring a special status

    Reward power – able to reward those who comply

Leadership: types of power

Personal power

  Expert power: holder can carry out specialist tasks that are in demand

  Information power: holder has access to needed information

  Referent power: based on personal attractiveness or charisma

Leadership styles

decision-making

	autocrat	democrat
directive		
permissive		

implementation

McGraw Hill **Education**

Leadership styles

Task orientation – focus on the work in hand

People orientation – focus on relationships

Where there is uncertainty about the way job is to be done or staff are inexperienced they welcome task oriented supervision

Uncertainty is reduced – people orientation more important

Risk that with reduction of uncertainty, managers have time on their hands and become more task oriented (interfering)