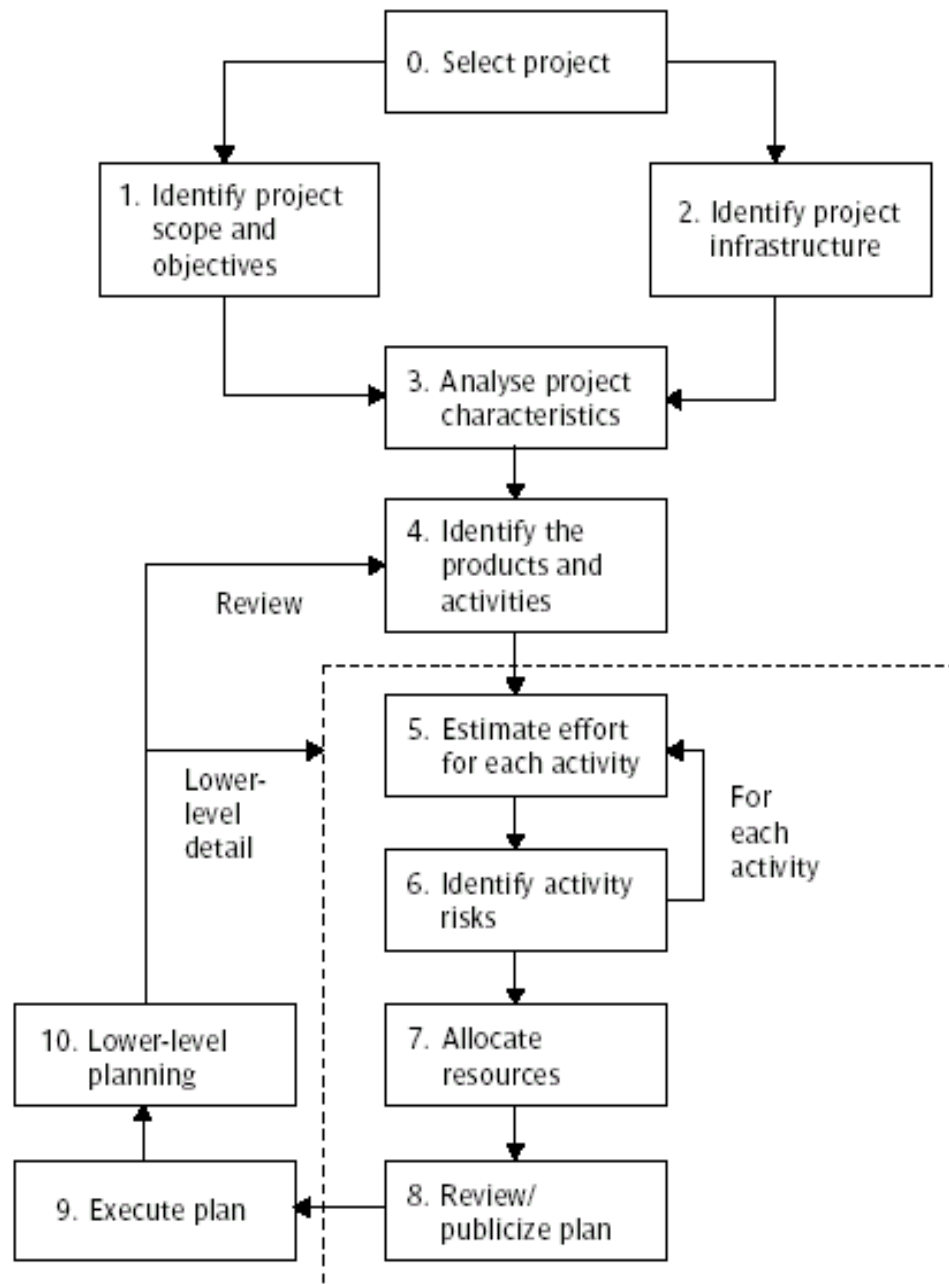*Step Wise Project planning*

Introduction to Step Wise Project Planning

- A framework of basic **steps in project planning**
- The framework described is called the Step Wise method to help to distinguish it from other methods such as PRINCE2.
- PRINCE2 is a set of project management standards that were originally sponsored by what is now the Office of Government Commerce (OGC) for use on British government ICT and business change projects.
- PRINCE stands for PRojects IN Controlled Environments.

An overview of Step Wise Project planning

0. Select project

1. Identify project scope and objectives

2. Identify project infrastructure

3. Analyse project characteristics

4. Identify the products and activities

Review

Lower-level detail

5. Estimate effort for each activity

6. Identify activity risks

For each activity

7. Allocate resources

10. Lower-level planning

9. Execute plan

8. Review/ publicize plan

Step	Activities within step
0	Select project
1	Identify project scope and objectives
1.1	Identify objectives and measures of effectiveness in meeting them
1.2	Establish a project authority
1.3	Identify stakeholders
1.4	Modify objectives in the light of stakeholder analysis
1.5	Establish methods of communication with all parties
2	Identify project infrastructure
2.1	Establish relationship between project and strategic planning
2.2	Identify installation standards and procedures
2.3	Identify project team organization

3   Analyse project characteristics

3.1   Distinguish the project as either objective- or product-driven

3.2   Analyse other project characteristics

3.3   Identify high-level project risks

3.4   Take into account user requirements concerning implementation

3.5   Select general life-cycle approach

3.6   Review overall resource estimates

4   Identify project products and activities

4.1   Identify and describe project products   (including quality criteria)

4.2   Document generic product flows

4.3   Recognize product instances

4.4   Produce ideal activity network

4.5   Modify ideal to take into account need for stages and checkpoints

5   Estimate effort for each activity

5.1   Carry out bottom-up estimates

5.2   Revise plan to create controllable activities

6   Identify activity risks

6.1   Identify and quantify activity-based risks

6.2   Plan risk reduction and contingency measures where appropriate

6.3   Adjust plans and estimates to take account of risks

7   Allocate resources

7.1   Identify and allocate resources

7.2   Revise plans and estimates to take account of resource constraints

8   Review/publicize plan

8.1   Review quality aspects of project plan

8.2   Document plans and obtain agreement

9/10   Execute plan/lower levels of planning

This may require the reiteration of the planning process at a lower level

Step 1: Identify Project Scope and Objectives

- Steps 1, 2 and 3 are longer-term planning, broad in outline

  - 1.1 Identify objectives and *practical* measures of effectiveness in meeting them

    - vision document helps find the objectives
    - measuring effectiveness can be in terms of *software quality...*

Software Quality

- *product operation quality factors:*
  - correctness - fulfils user objectives / meet specifications
  - reliability - failure rate / degree of accuracy
  - efficiency - computer resources required
  - integrity - safekeeping of data
  - usability - effort required to learn and use
- *product revision quality factors:*
  - maintainability - effort required to locate and fix errors
  - testability - effort required to test / scope, precision of test
  - flexibility - effort required to modify
- *product transition quality factors:*
  - portability - effort required to switch hardware/OS
  - reusability - of components in other applications
  - interoperability - effort required to couple to another system

Step 1: Identify Project Scope and Objectives

- 1.2 Establish a project authority
  - a single person or group with unity of purpose
  - to avoid being pulled in different directions

- 1.3 Stakeholder analysis - identify all stakeholders in the project and their interests &
- 1.4 Modify objectives in light of the stakeholder analysis
  - again, can look to the vision document

- 1.5 Establish methods of communication with all parties
  - including external authorities/providers
  - might lead to making a *communications plan*

Step 2: Identify Project Infrastructure

- 2.1 Identify relationship between the project and strategic planning
  - a strategic business or it plan needs to document:
  - order of projects
  - hardware & software standards to be met

- 2.2 Identify installation standards and procedures
  - making sure that all changes are documented, approved and reviewed
  - specifying which measure of quality are to be used and when

- 2.3 Identify project team organisation
  - can you choose, or is it pre-specified?
  - either way, what impact will team and sub-team structure have?

Step 3: Analyse Project Characteristics

- 3.1 Distinguish the project as either objective- or product-driven

  - objective driven will give you more freedom but often there is a specified product you have to build to form a solution

- 3.2 Analyse other project characteristics (including quality-based ones)

  - essentially considering non-functional requirements:

    - safety critical?

    - sensitive data?

    - speed/space requirements?

Step 3: Analyse Project Characteristics

- 3.3 Identify high level project risks

  - as discussed in the Iteration Planning lecture

  - don't forget aspects such as resistance to change

- 3.4 Take into account user requirements concerning implementation

  - some organisations (such as government) might require use of the waterfall method

Step 3: Analyse Project Characteristics

- if 3.4 allows a choice, then figure what's best based upon:

  - staff available

  - time available
  - distribution of resources

3.5 Select development methodology and lifecycle approach

- 3.6 Review overall resource estimates

  - and if need be revise cost estimates, team organisation and risks

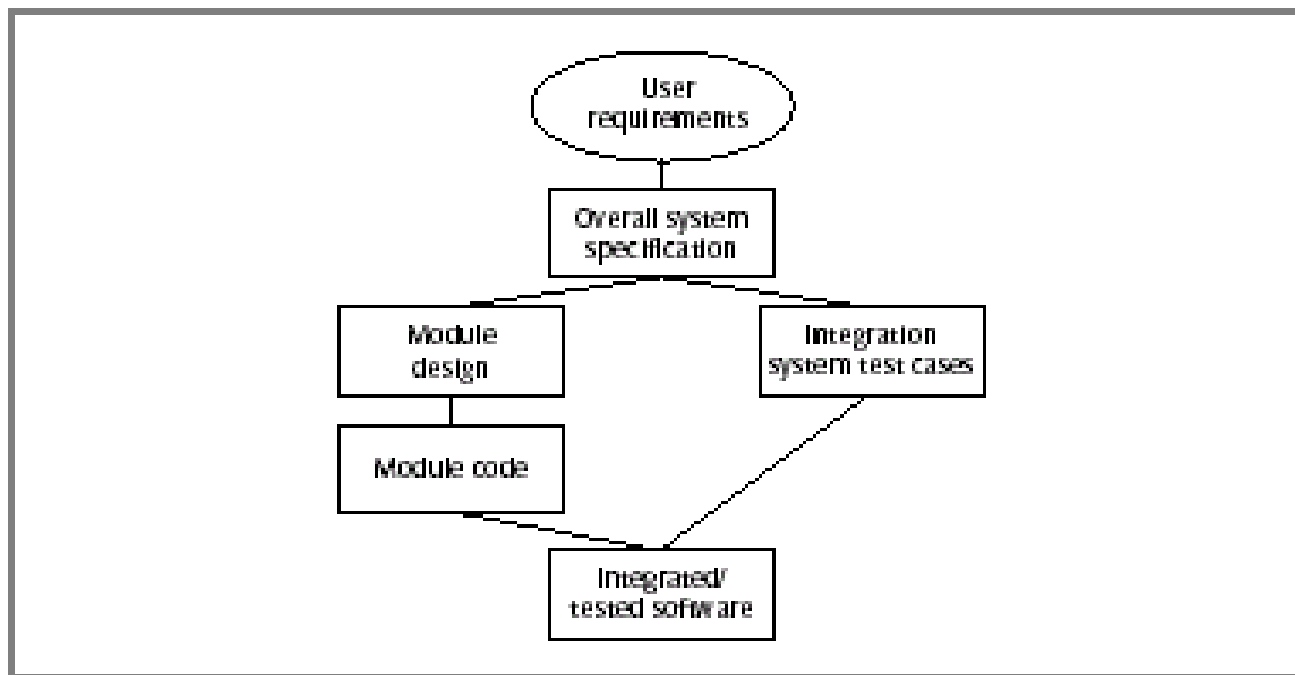# Step 4: Identify Project Products and Activities

- more detailed planning of individual activities

- 4.1 Identify and describe project products
  - activities should produce tangible products:
    - *deliverables* - to be hand over to the client at the end of the project
      - also includes technical products:
        - training manual, operating instructions
    - *intermediates* - used in the process of creating deliverables
      - also includes planning and quality products:
        - uml documentation, test cases

Products

- products can be composite - made up of several smaller (sub) products
- each should be documented by a *product description* (PRINCE2):
  - name
  - purpose
  - derivation (if it modifies an existing product)
  - composition
  - form
  - relevant standards
  - quality criteria (for acceptance)

Step 4: Identify Project Products and Activities

- 4.2 Document generic product flows

  - some products cannot be created until other products exist

  - these relationships can be captured in a *product flow diagram:*

A fragment of a Product Flow Diagram (PFD) for a software
development task

Step 4: Identify Project Products and Activities

- 4.3 Recognise product instances

  - spot when the same PFD fragment relates to many instances of a type product
  - that might allow you to

    - re-use the plans for the activities which produce it

    - assign team members to undertake groups of activities with similar pfd

Step 4: Identify Project Products and Activities
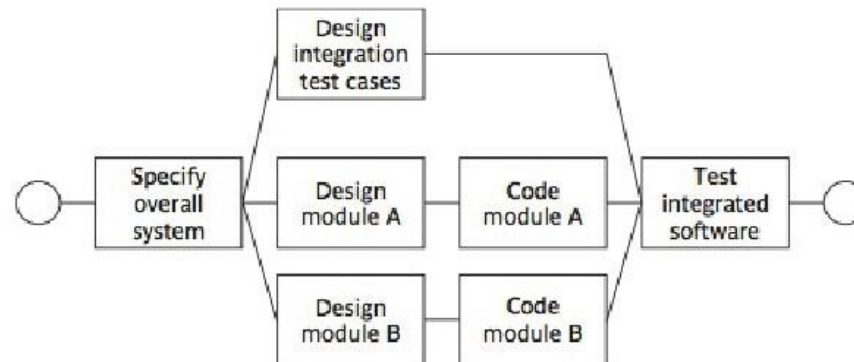
- 4.4 Produce ideal activity network

fig3.5 Ch 3 *Software Project Management*

- 'ideal' because resource constraints are not taken into account

Step 4: Identify Project Products and Activities

- 4.5 Modify the ideal to take into account need for stages and checkpoints

  - sequencing activities in 4.4 encourages a plan which will minimise the project time

  - it assumes that a dependent activity can start as soon as the preceeding ones have completed

  - in reality we will want to divide the project into stages and introduce *checkpoint activities*...

Checkpoint Activities & Milestones

- checking that products are compatible

- just after a checkpoint is a good place to add a *milestone:*

  - a dummy activity with no duration

  - indicates the start or end of a group of activities

  - can represent the completion of an important stage of a project

  - so useful for ensuring that overall monitoring of the project (such as by the project authority) takes place regularly at appropriate points

Step 5: Estimate effort for each activity

- 5.1 Carry out bottom-up estimates

  - staff, time, effort (staff x time), other resources needed

- 5.2 Revise plan to create controllable activities

  - long activities (say 12 weeks) make a project difficult to control

    - after 6 weeks are we 50% complete?

    - can be hard to tell

  - better to break down into smaller subtasks

  - conversely, some very short, connected activities might be better bundled together, with a simple checklist

  - roughly aim for activities to match the length of the reporting period

    - if you have progress meetings every 2 weeks, try to identify activities which take two weeks

Step 6: Identify activity risks

- 6.1 Identify and quantify activity-based risks
    - look at the assumptions in the plan, such as:
        - time required
        - availability of staff/resources
    - these generate uncertainty
        - simple way to handle:
            - create a most likely estimate for time/effort
            - create a second estimate with a safety margin such that the target has a 95% chance of being met
    - look at the damage that could be caused by a risk
        - pick out the most important ones

Step 6: Identify activity risks

- 6.2 Plan risk reduction and contingency measures where appropriate

  - reduce where possible

  - otherwise specify a *contingency plan*

    - for example: contract temporary developer if team member becomes unavailable through illness

- 6.3 Adjust overall plans and estimates to take account of risks

  - including adding new activities - such as training and practice - if need be

Step 7: Allocate Resources

- 7.1 Identify and allocate resources

  - what type of staff is needed for activity?

  - who is (provisionally) available when required?

- 7.2 Revise plans and estimates to take into account resource constraints

  - where there is conflict establish an order of priority

  - note effects upon project duration

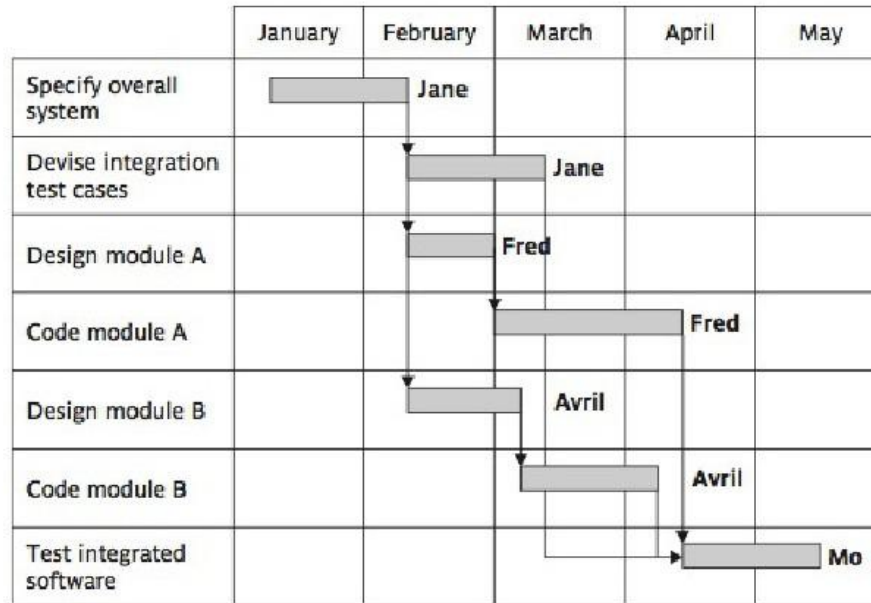 - a *GANNT* chart can help resolve conflict and maximise productivity...

GANNT Chart

	January	February	March	April	May
Specify overall system	▭	Jane			
Devise integration test cases		▭ Jane			
Design module A		▭ Fred			
Code module A			▭ Fred		
Design module B		▭ Avril			
Code module B			▭ Avril		
Test integrated software				▭ Mo	

fig3.6 Ch 3 *Software Project Management (5th Edition, 2009) Hughes & Cotterell*

Step 8: Review/Publicise plan

- 8.1 Review quality aspects of the project plan

    - sometimes undertaking one activity can reveal that an earlier activity was not properly completed:

        - will have to be reworked

        - will require effort and resources

        - can lead to loss of control of project

    - need to be sure that a completed task is truly completed

        - need *quality criteria* for each task

            - tick off when complete

                - the list from step 1.1 will help form these

Step 8: Review/Publicise plan

- 8.2 Document plans and obtain agreement

    - make sure everyone understands and agrees

    - specify this task in a *communications plan* if need be (as mentioned in step 1.5)

Steps 9 and 10: Execute Plan / Lower Levels of Planning

- during the project draw up plans for activities in greater detail as they become due

    - detail has to wait as more information becomes available

    - especially if you are using an iterative development approach

    - maintain provisional plans for more important later tasks

    - planning in great detail too soon could be a waste of time

To conclude

- planning a project properly is almost a project in itself

    - the Step Wise method is one good approach

- documenting the plan is important:

    - the activities

    - the products

    - the schedule

    - the measures of quality

    - the lines of communication

    - the procedures/standards which must be met

- this is a vast topic and wider reading for those interested is recommended

References

- Hughes, B., and Cotterell, M. (1999) *Software Project Management*, 2nd edition, McGraw-Hill.

- PMBOK, 6TH Edition

- Pfleeger, S.L. (1998) *Software Engineering: Theory and Practice*, Prentice Hall.

Next … Project Scope Management