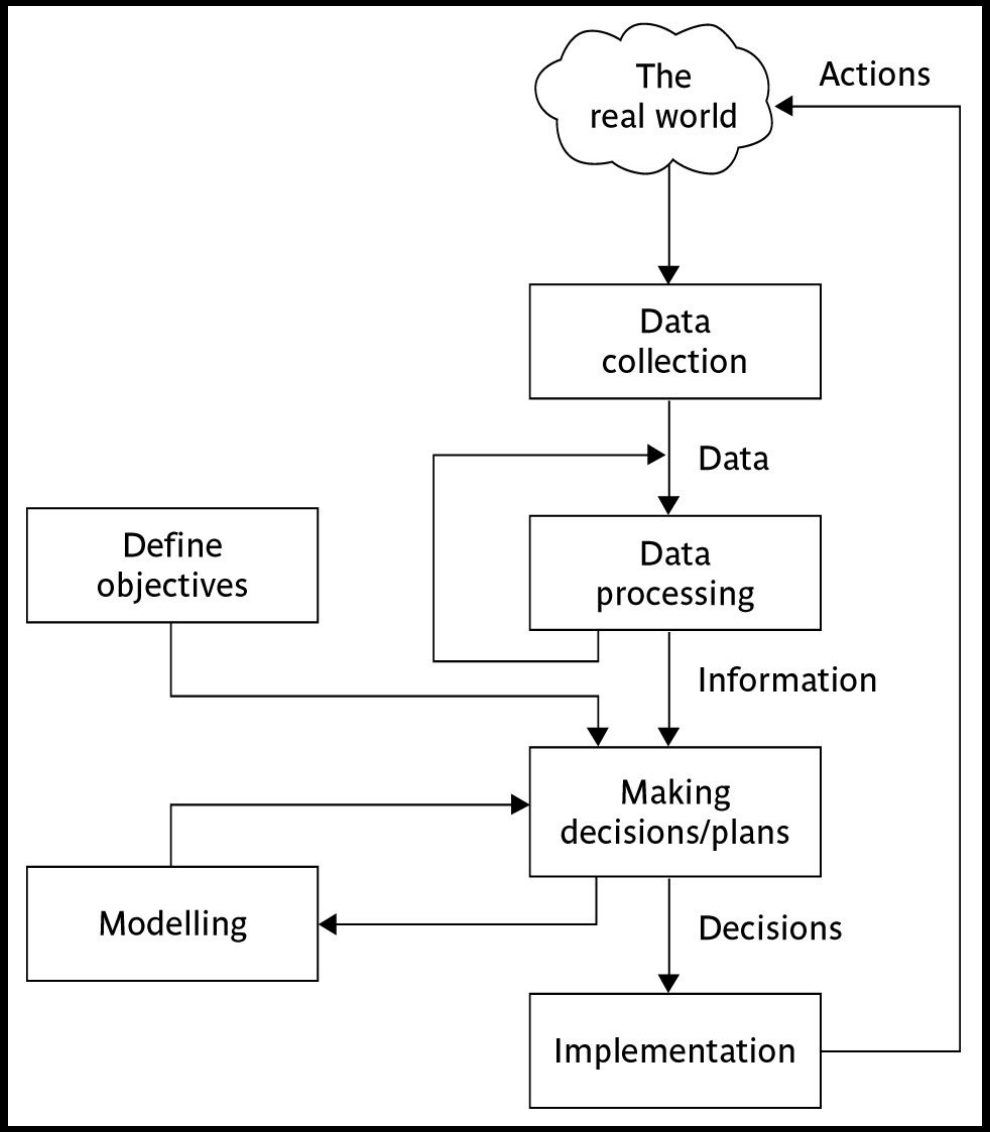
Software Project Management
Fifth Edition

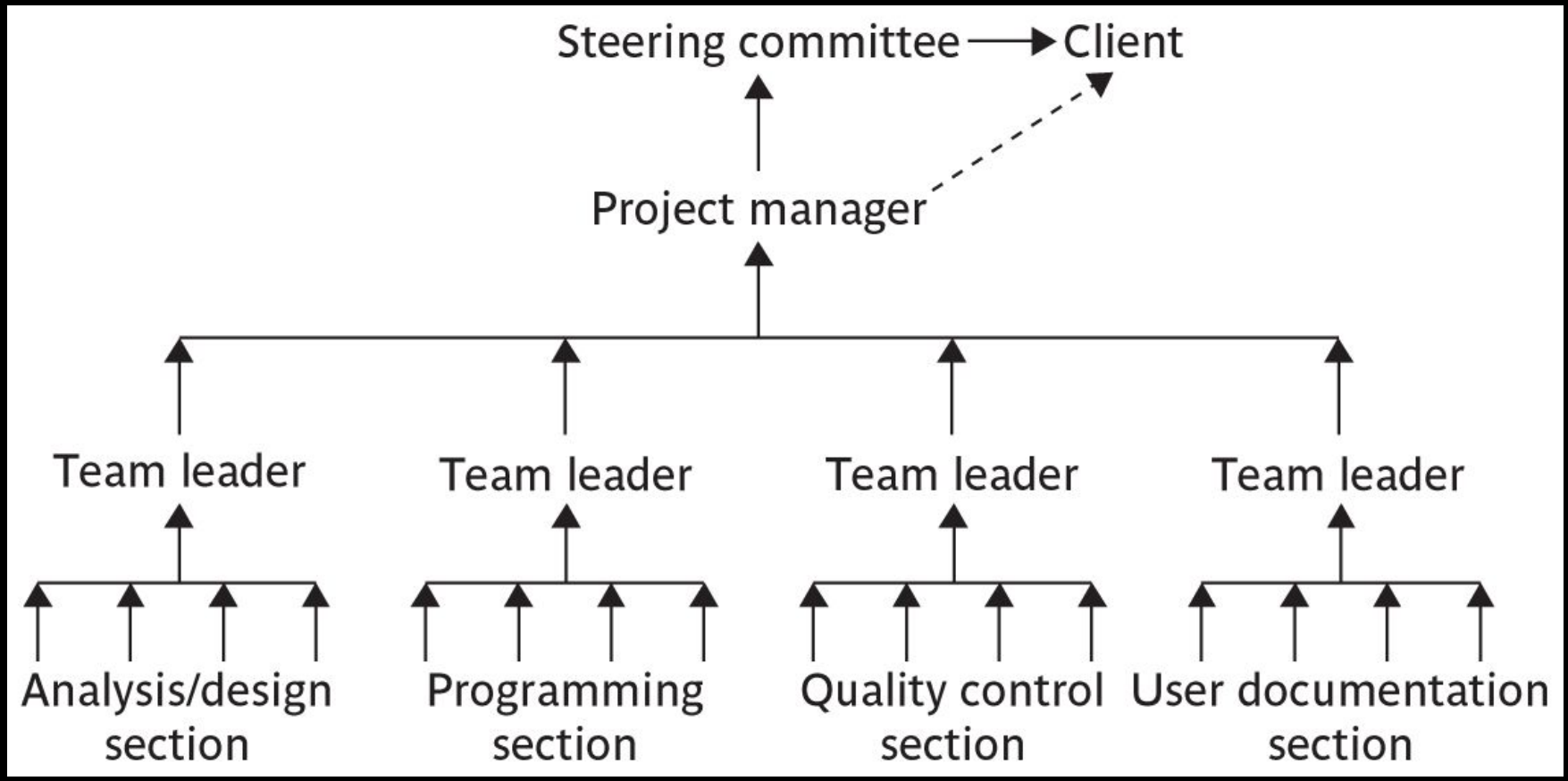
Chapter 9

Monitoring and control

The control cycle



Responsibilities



Assessing progress

- Checkpoints – predetermined times when progress is checked
  - Event driven: check takes place when a particular event has been achieved
  - Time driven: date of the check is pre-determined

- Frequency of reporting
  - The higher the management level then generally the longer the gaps between checkpoints

Collecting progress details

Need to collect data about:

- Achievements
- Costs

A big problem: how to deal with *partial completions*

99% completion syndrome

Possible solutions:

- Control of products, not activities
- Subdivide into lots of sub-activities

Red/Amber/Green reporting

Identify key tasks

Break down into sub-tasks

Assess subtasks as:

- Green – ‘on target’
- Amber – ‘not on target but recoverable’
- Red – ‘not on target and recoverable only with difficulty’

Status of ‘critical’ tasks is particularly important

Review

Review of work products is an important mechanism for monitoring the progress of a project and ensuring the quality of the work products.

Testing is an effective defect removal mechanism.

- However, testing is applicable to only executable code.
- Review is applicable to all work products.

Utility of Review

A cost-effective defect removal mechanism.

Review usually helps to identify any deviation from standards.

Reviewers suggest ways to improve the work product

a review meeting often provides learning opportunities to not only the author of a work product, but also the other participants of the review meeting.

The review participants gain a good understanding of the work product under review, making it easier for them to interface or use the work product in their work.

Review Roles

Moderator:

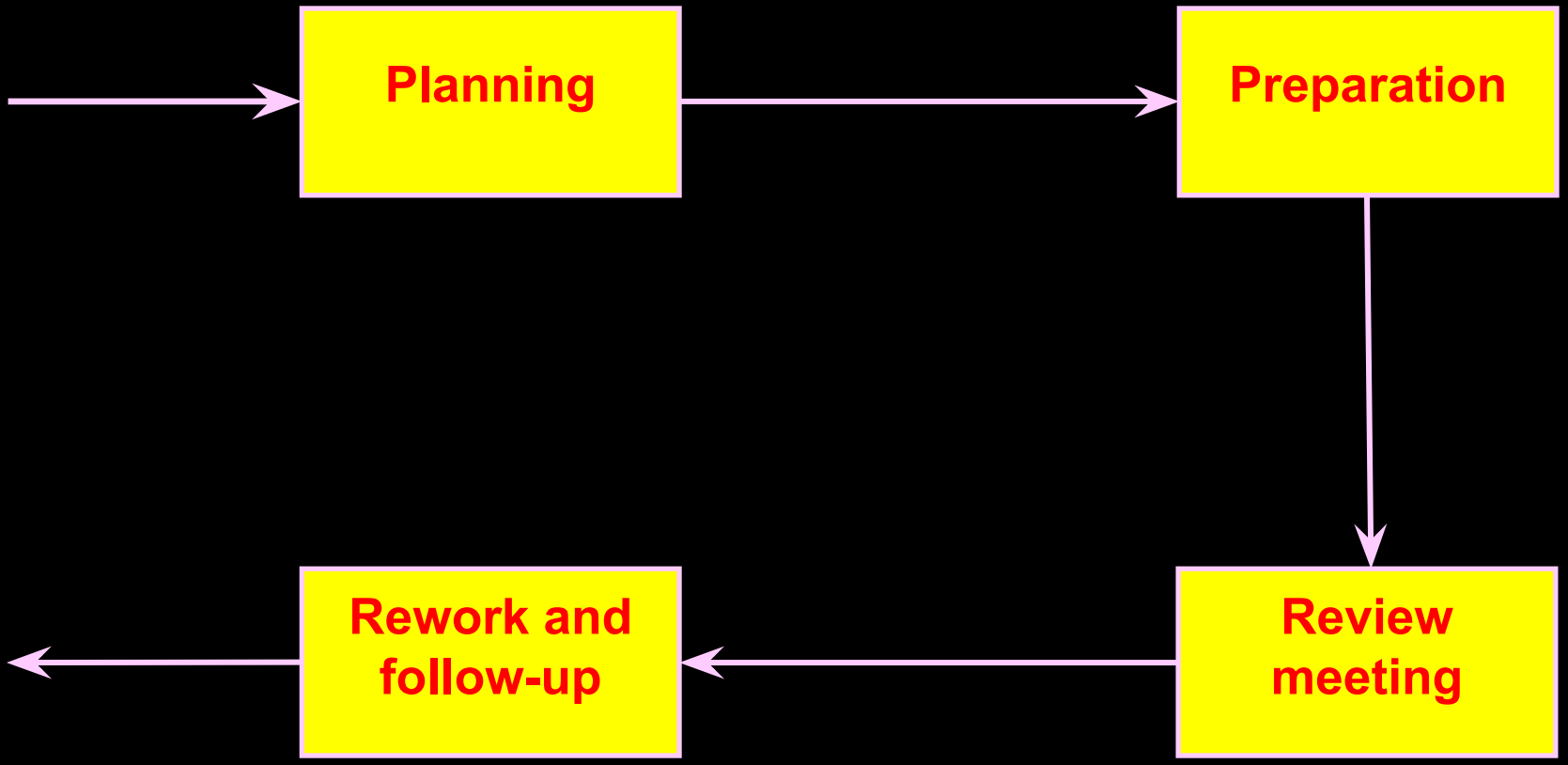
- Schedules and convenes meetings, distributes review materials, leads and moderates review sessions.

Recorder:

- Records the defects found and the time and effort data.

Reviewers.

Review Process



Project Termination Review

Project termination reviews provide important opportunities to learn from past mistakes as well as successes.

Project termination need not necessarily mean project failure or premature abandonment.

- A project may be terminated on successful completion

Reasons for Project Termination

Project is completed successfully handed over to the customer.

Incomplete requirements

Lack of resources

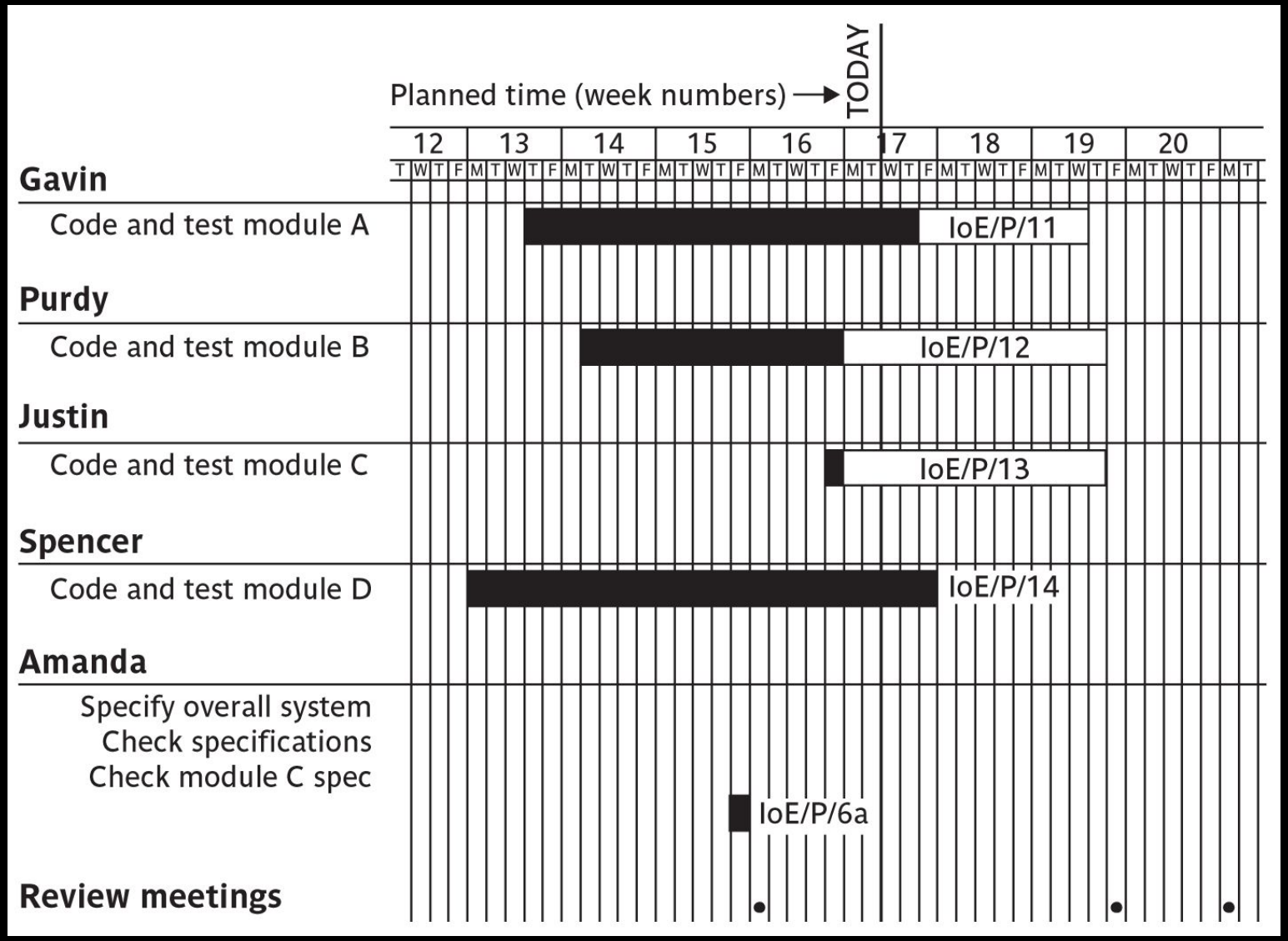
Some key technologies used in the project have become obsolete during project execution

Economics of the project has changed, for example because many competing product may have become available in the market.

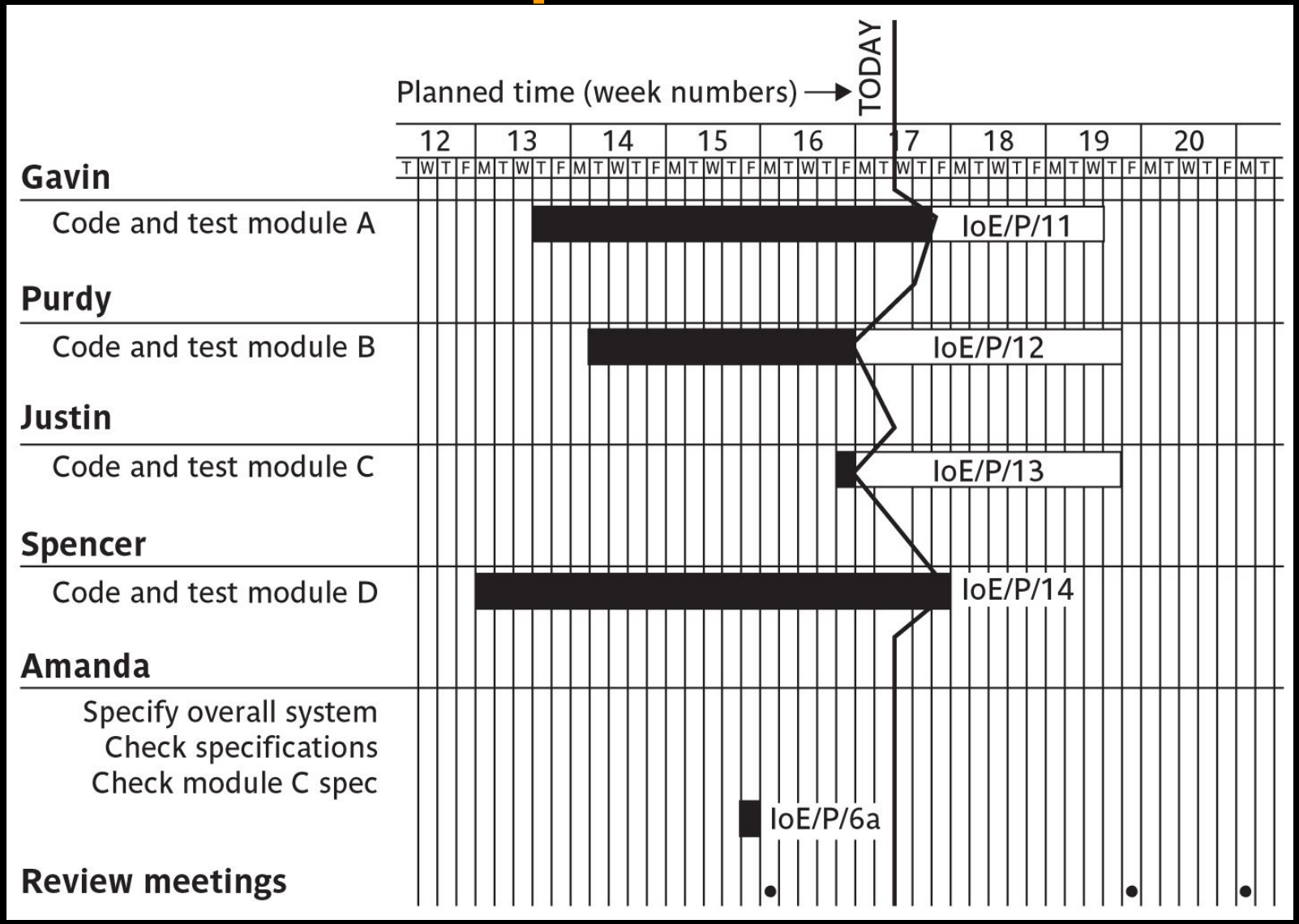
Project Termination Process

- Project survey
- Collection of objective information
- Debriefing meeting
- Final project review
- Result publication

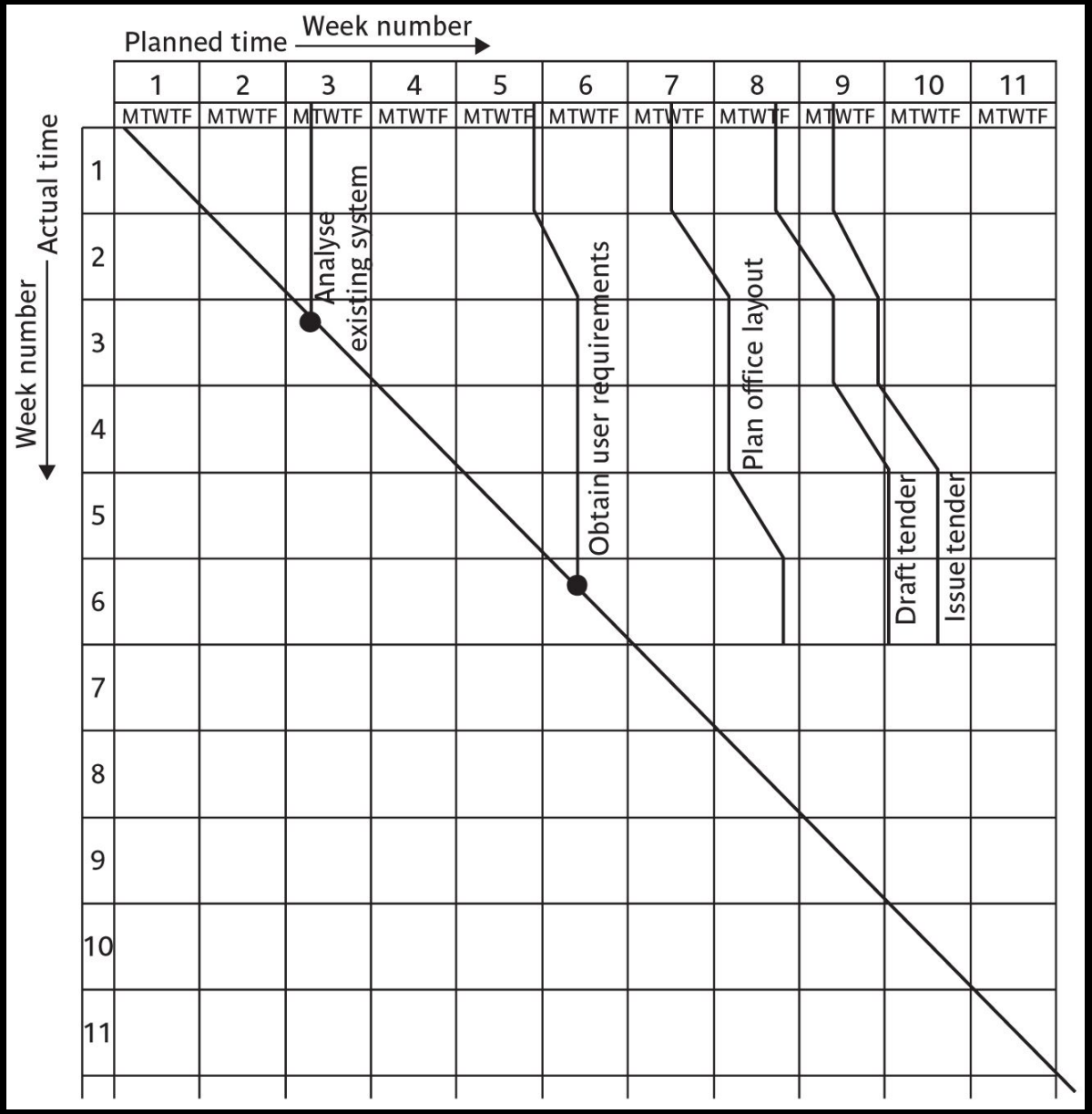
Gantt charts



Slip charts


Planned time (week numbers)
TODAY
12
13
14
15
16
17
18
19
20
T W T F M T W T F M T W T F M T W T F M T W T F M T W T F M T W T F M T W T F M T W T F M T W T F M T
Gavin
Code and test module A
IoE/P/11
Purdy
Code and test module B
IoE/P/12
Justin
Code and test module C
IoE/P/13
Spencer
Code and test module D
IoE/P/14
Amanda
Specify overall system
Check specifications
Check module C spec
IoE/P/6a
Review meetings

The timeline



Cost monitoring

A project could be late because the staff originally committed have not been deployed

In this case the project will be *behind time* but *under budget*

A project could be on time but only because additional resources have been added and so be *over budget*

Need to monitor both achievements and costs

Earned value analysis

Planned value (PV) or *Budgeted cost of work scheduled (BCWS)* – original estimate of the effort/cost to complete a task (compare with idea of a 'price')

Earned value (EV) or *Budgeted cost of work performed* (BCWP) – total of PVs for the work completed at this time

Accounting conventions

Work completed allocated on the basis

- *50/50* half allocated at start, the other half on completion. These proportions can vary e.g. 0/100, 75/25 etc
- *Milestone* current value depends on the milestones achieved
- *Units processed*

Can use money values, or staff effort as a surrogate

Earned value – an example

Tasks

Specify module	5 days
Code module	8 days
Test module	6 days

At the beginning of day 20, PV = 19 days

If everything but testing completed EV = 13 days

Schedule variance = EV-PV i.e. 13-19 = -6

Schedule performance indicator (SPI) = 13/19 = 0.68

SV negative or SPI <1.00, project behind schedule

Earned value analysis – actual cost

Actual cost (AC) is also known as Actual cost of work performed (ACWP)

In previous example, if

- ‘Specify module’ actually took 3 days
- ‘Code module’ actually took 4 days

Actual cost = 7 days

Cost variance (CV) = EV-AC i.e. 13-7 = 6 days

Cost performance indicator = 13/7 = 1.86

Positive CV or CPI > 1.00 means project within budget

Earned value analysis – actual costs

CPI can be used to produce new cost estimate

Budget at completion (BAC) – current budget allocated to total costs of project

Estimate at completion (EAC) – updated estimate = BAC/CPI

e.g. say budget at completion is £19,000 and CPI is 1.86

EAC = BAC/CPI = £10,215 (projected costs reduced because work being completed in less time)

Time variance

Time variance (TV) – difference between time when specified EV should have been reached and time it actually was

For example say an EV of £19000 was supposed to have been reached on 1st April and it was actually reached on 1st July then TV = - 3 months

Earned value chart with revised forecasts

Activity Assessment Sheet

Staff Justin

Ref: IoE/P/13 Activity: Code and test module C

Week number	13	14	15	16	17	18	
Activity summary	G	A	A	R			
Component							Comments
Screen handling procedures	G	A	A	G			
File update procedures	G	G	R	A			
Housekeeping procedures	G	G	G	A			
Compilation	G	G	G	R			
Test data runs	G	G	G	A			
Program documentation	G	G	A	R			

Prioritizing monitoring

We might focus more on monitoring certain types of activity e.g.

- Critical path activities
- Activities with no free float – if delayed later dependent activities are delayed
- Activities with less than a specified float
- High risk activities
- Activities using critical resources

Getting back on track: options

Renegotiate the deadline – if not possible then

Try to shorten critical path e.g.

- Work overtime
- Re-allocate staff from less pressing work
- Buy in more staff

Reconsider activity dependencies

- Over-lap the activities so that the start of one activity does not have to wait for completion of another
- Split activities

Exception planning

Some changes could affect

- Users
- The business case (e.g. costs increase reducing the potential profits of delivered software product)

These changes could be to

- Delivery date
- Scope
- Cost

In these cases an **exception report** is needed

Exception planning - continued

First stage

Write an **exception report** for sponsors (perhaps through project board)

- Explaining problems
- Setting out options for resolution

Second stage

Sponsor selects an option (or identifies another option)

Project manager produces an **exception plan** implementing selected option

Exception plan is reviewed and accepted/rejected by sponsors/Project Board

Change control

The role of configuration librarian:

- Identifying items that need to be subject to change control
- Management of a central repository of the master copies of software and documentation
- Administering change procedures
- Maintenance of access records

Typical change control process

1. One or more users might perceive the need for a change
2. User management decide that the change is valid and worthwhile and pass it to development management
3. A developer is assigned to assess the practicality and cost of making the change
4. Development management report back to user management on the cost of the change; user management decide whether to go ahead

Change control process contd.

5. One or more developers are authorized to make copies of components to be modified
6. Copies modified. After initial testing, a test version might be released to users for acceptance testing
7. When users are satisfied then operational release authorized – master configuration items updated

Software Configuration Management (SCM)

SCM is concerned with tracking and controlling changes to a software.

Development and maintenance environment:

- Various work products associated with the software continually change.
- Unless a proper configuration management system is deployed, several problems can appear.

Why Use SCM?

- Problems associated with concurrent access
- Undoing Changes
- System accounting
- Handling variants
- Accurate determination project status
- Preventing unauthorized access to the work products

Configuration Control

Two main operations:

- Reserve
- Restore