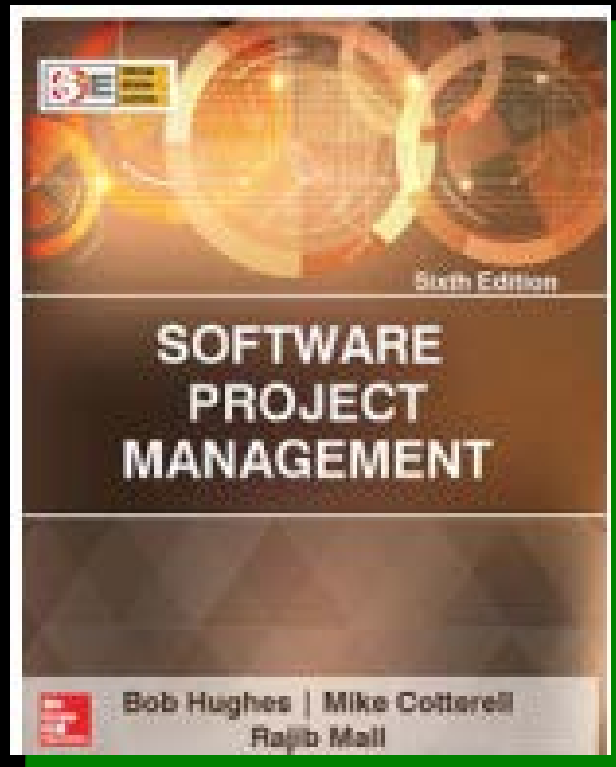


Software Project Management Sixth Edition



Chapter 8

Resource allocation

Schedules

- *Activity schedule* - indicating start and completion dates for each activity
- *Resource schedule* - indicating dates when resources needed + level of resources
- *Cost schedule* showing accumulative expenditure

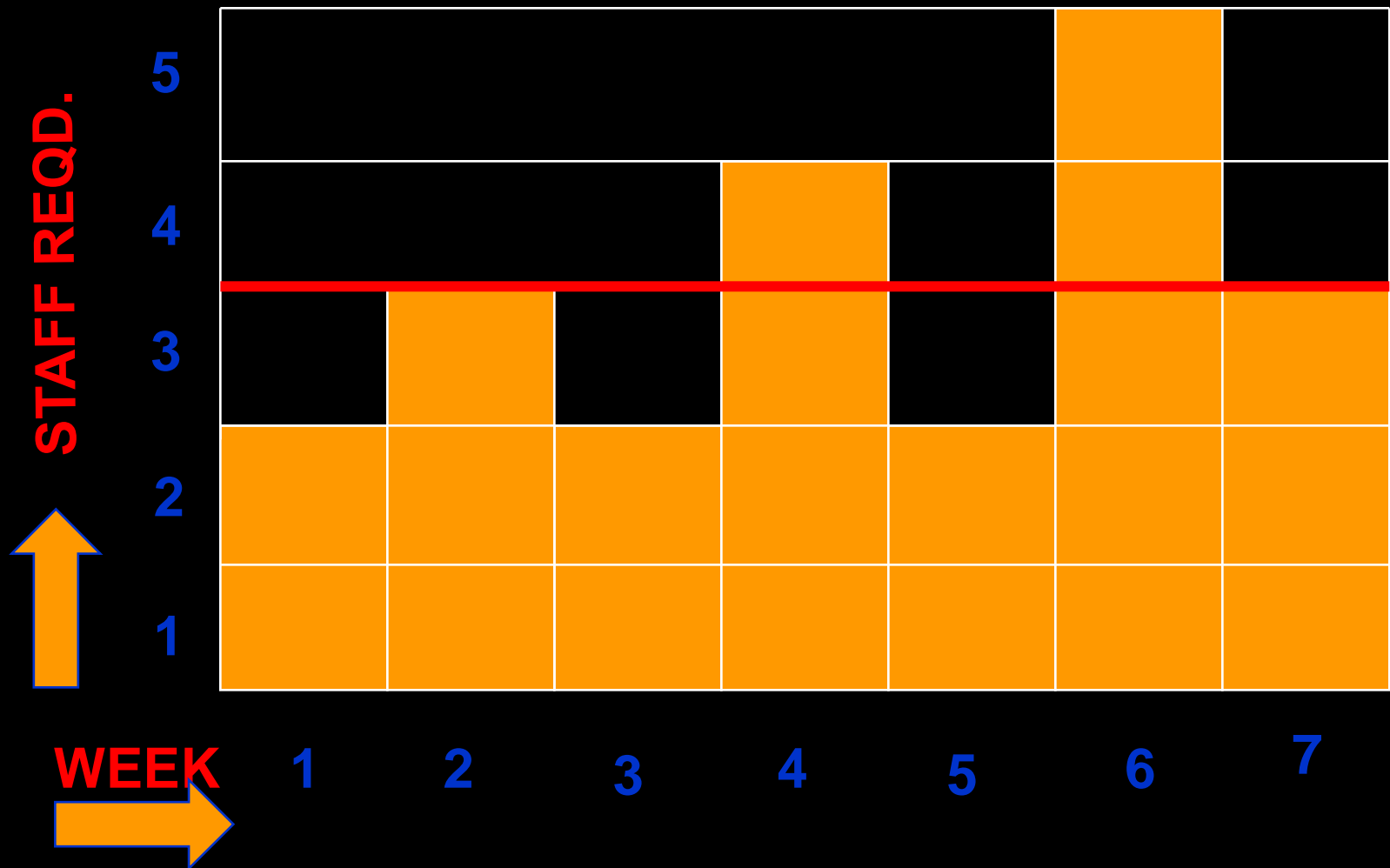
Resources

- These include
 - ◆ labour
 - ◆ equipment (e.g. workstations)
 - ◆ materials
 - ◆ space
 - ◆ services
- Time: elapsed time can often be reduced by adding more staff
- Money: used to buy the other resources

Resource allocation

- Identify the resources needed for each activity and create a *resource requirement list*
- Identify *resource types* - individuals are interchangeable within the group (e.g. 'VB programmers' as opposed to 'software developers')
- Allocate resource types to activities and examine the *resource histogram*

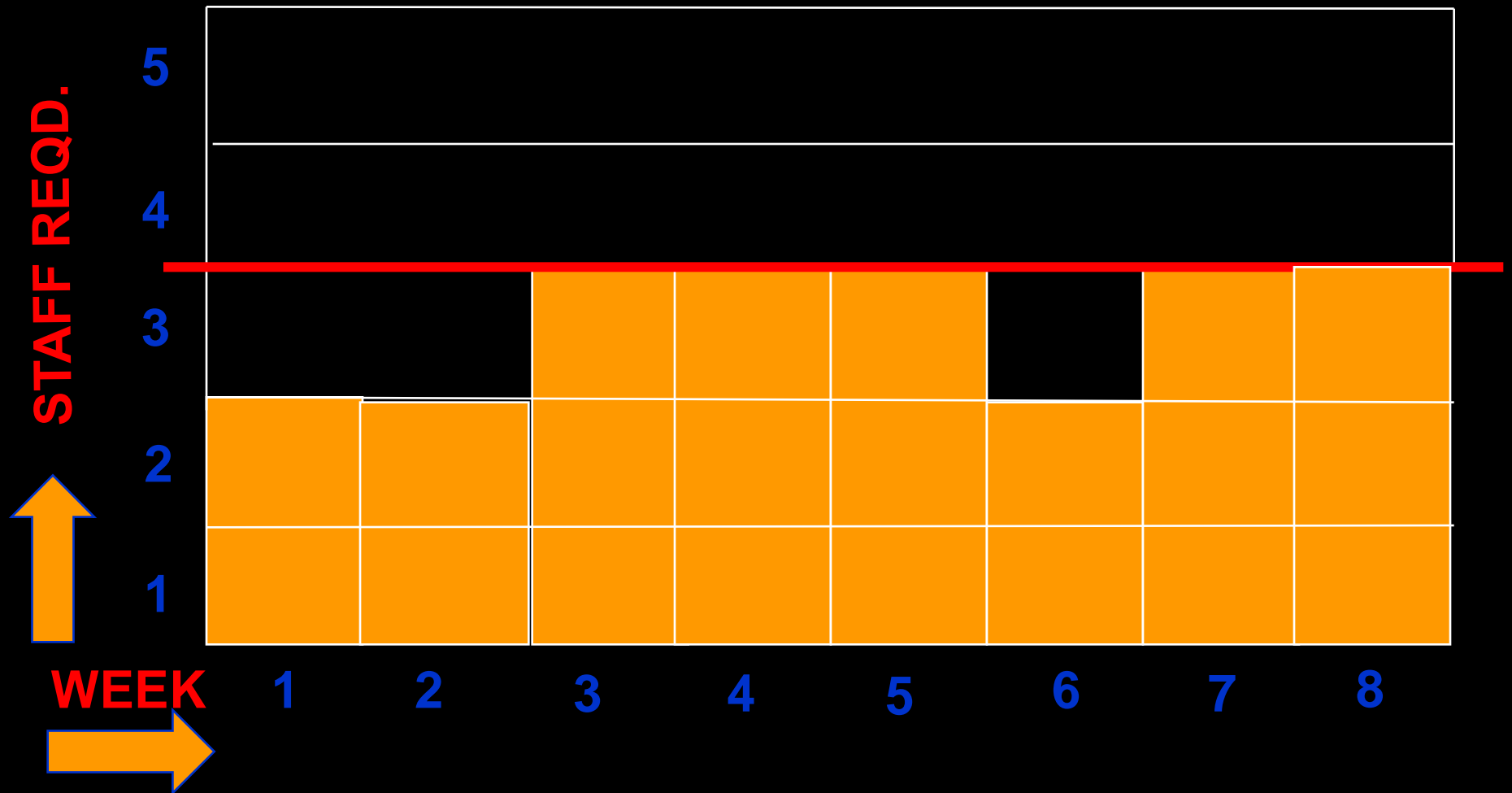
Resource histogram: systems analysts



Resource smoothing

- It is usually difficult to get specialist staff who will work odd days to fill in gaps – need for staff to learn about application etc
- Staff often have to be employed for a continuous block of time
- Therefore desirable to employ a constant number of staff on a project – who as far as possible are fully employed
- Hence need for **resource smoothing**

Resource smoothing



Resource clashes

- Where same resource needed in more than one place at the same time
- can be resolved by:
 - ◆ delaying one of the activities
 - taking advantage of float to change start date
 - delaying start of one activity until finish of the other activity that resource is being used on - *puts back project completion*
 - ◆ moving resource from a non-critical activity
 - ◆ bringing in additional resource - *increases costs*

Prioritizing activities

There are two main ways of doing this:

- *Total float priority* – those with the smallest float have the highest priority
- *Ordered list priority* – this takes account of the duration of the activity as well as the float – see next overhead

Burman's priority list

Give priority to:

- Shortest critical activities
- Other critical activities
- Shortest non-critical activities
- Non-critical activities with least float
- Non-critical activities

Resource usage

- Need to maximise %usage of resources i.e. reduce idle periods between tasks
- Need to balance costs against early completion date
- Need to allow for contingency

Critical path

- Scheduling resources can create new dependencies between activities – recall *critical chains*
- It is best not to add dependencies to the activity network to reflect resource constraints
 - ◆ Makes network very messy
 - ◆ A resource constraint may disappear during the project, but link remains on network
- Amend dates on **schedule** to reflect resource constraints

Allocating individuals to activities

The initial 'resource types' for a task have to be replaced by actual individuals.

Factors to be considered:

- Availability
- Criticality
- Risk
- Training
- Team building – and motivation

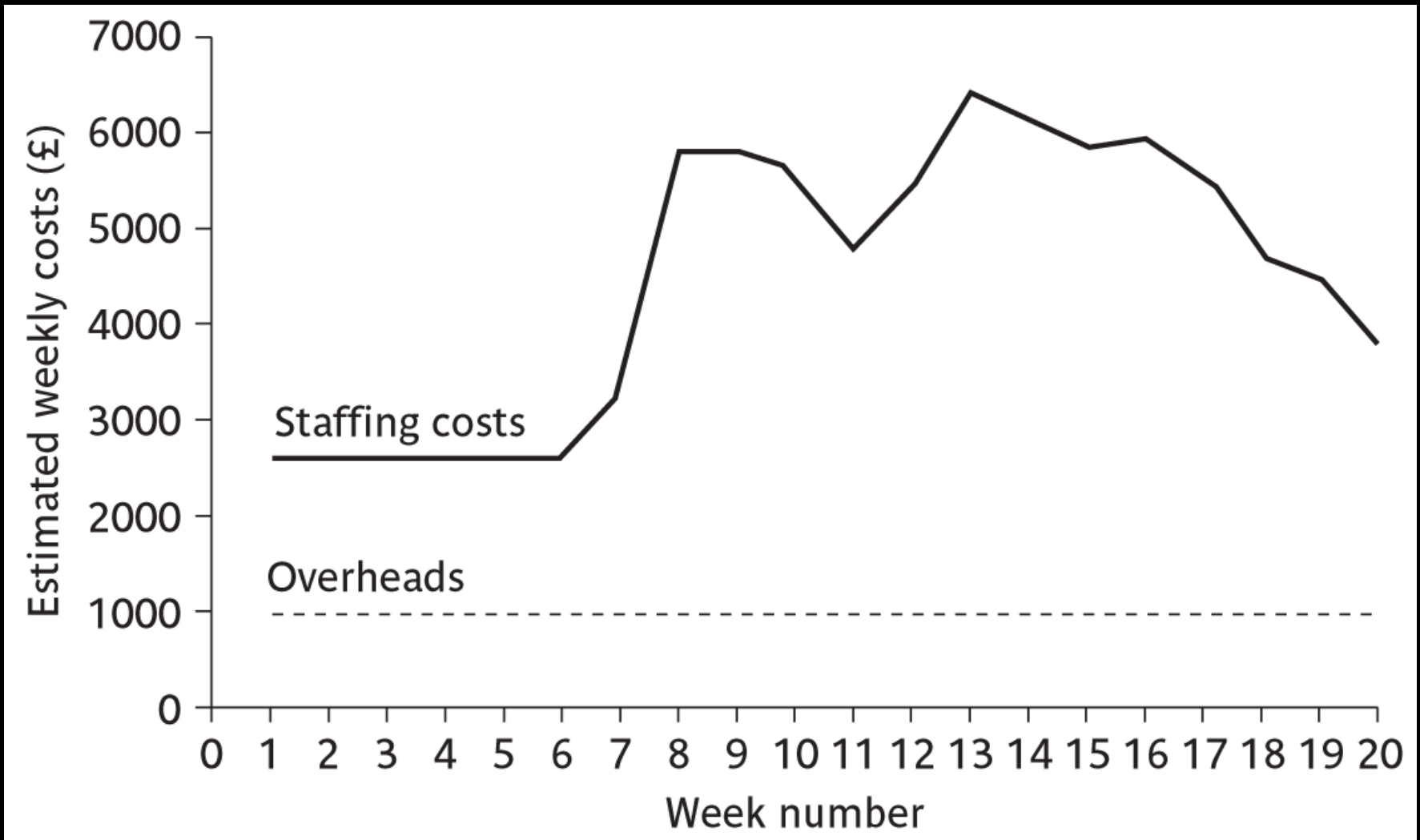
Cost schedules

Cost schedules can now be produced:

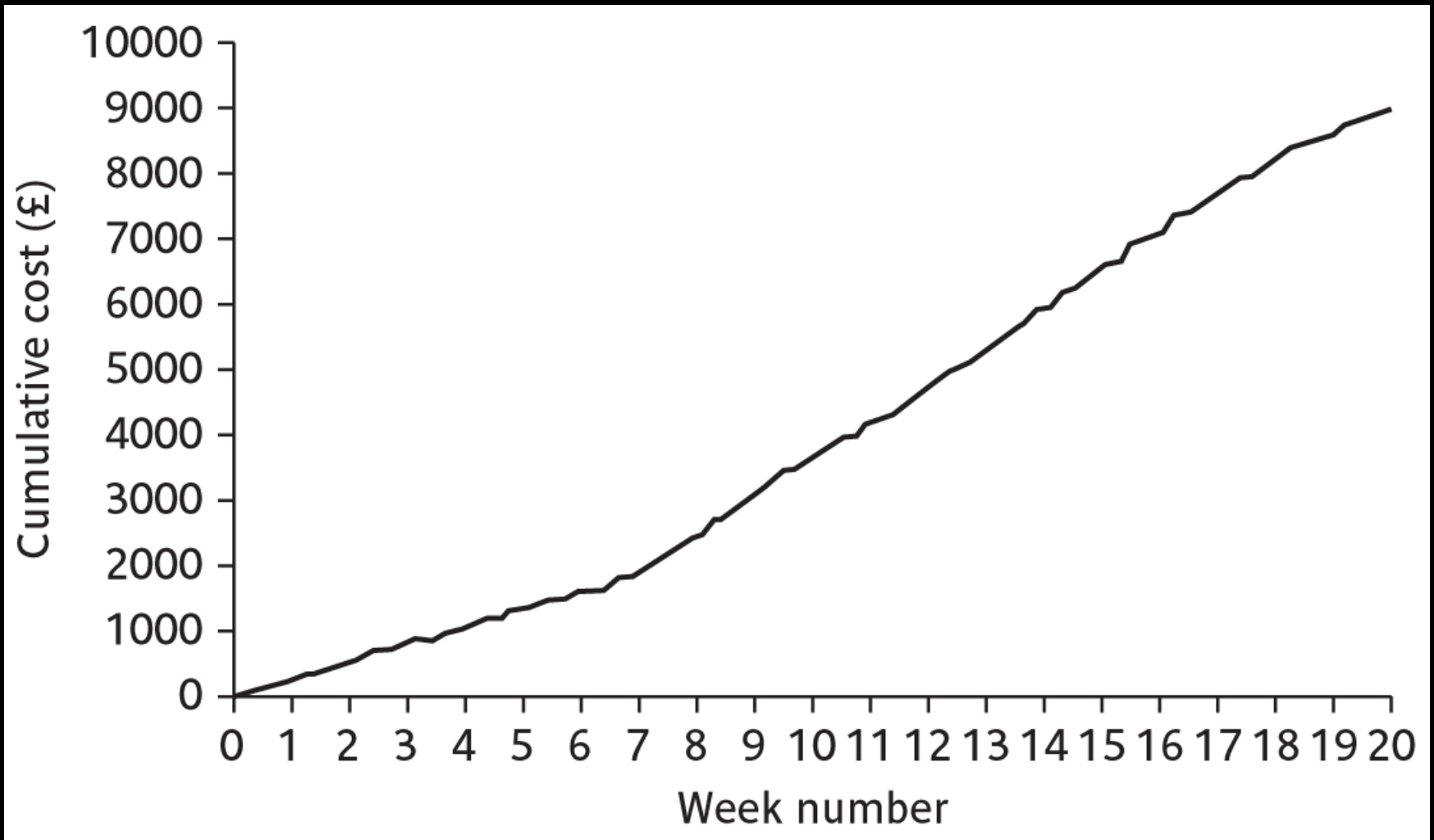
Costs include:

- Staff costs
- Overheads
- Usage charges

Cost profile



Accumulative costs



Balancing concerns

