Managing Contracts and People

Acquiring software from external supplier

This could be:

- a *tailored system* - created specially for the customer
- *off-the-shelf* - bought 'as is'
- *customised off-the-shelf* (COTS) - a core system is customised to meet needs of a particular customer

Types of contract

- fixed price contracts
- time and materials contracts
- fixed price per delivered unit

Note difference between goods and services

Often licence to use software is bought rather than the software itself

Fixed price contracts

Advantages to customer

- known expenditure
- supplier motivated to be cost-effective

Fixed price contracts

Disadvantages

- supplier will increase price to meet contingencies
- difficult to modify requirements
- cost of changes likely to be higher
- threat to system quality

Time and materials

Advantages to customer

- easy to change requirements
- lack of price pressure can assist product quality

Time and materials

Disadvantages

- Customer liability - the customer absorbs all the risk associated with poorly defined or changing requirements
- Lack of incentive for supplier to be cost-effective

Fixed price/unit

Advantages for customer

- customer understanding of how price is calculated
- comparability between different pricing schedules
- emerging functionality can be accounted for
- supplier incentive to be cost-effective

Fixed price/unit

Disadvantages

- difficulties with software size measurement - may need independent FP counter
- changing (as opposed to new) requirements: how do you charge?

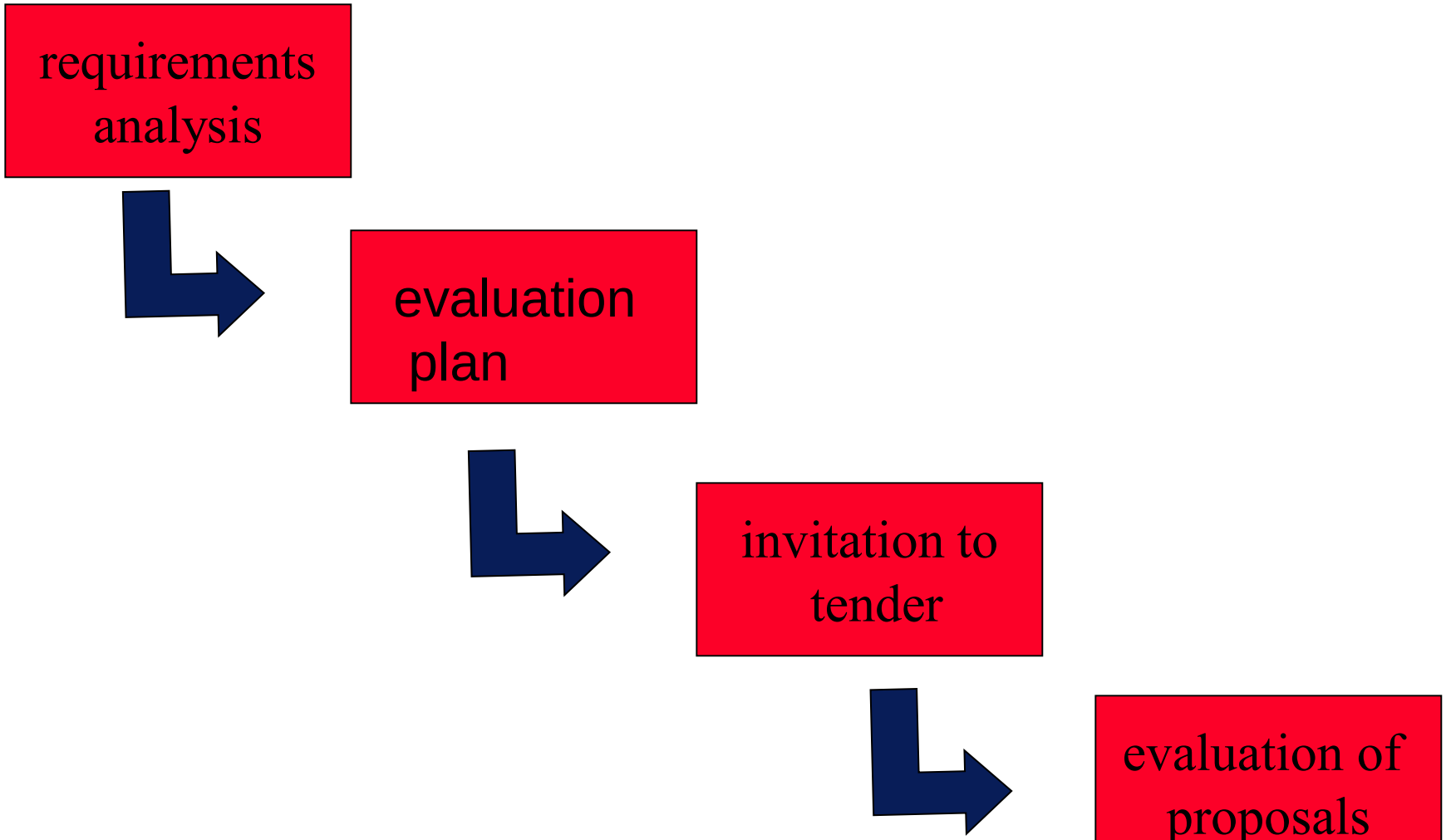
The tendering process

- Open tendering
  - any supplier can bid in response to the *invitation to tender*
  - all tenders must be evaluated in the same way
  - government bodies may have to do this by local/international law (including EU and WTO, World Trade Organization, requirements

The tendering process

- Restricted tendering process
  - bids only from those specifically invited
  - can reduce suppliers being considered at any stage
- Negotiated procedure
  - negotiate with one supplier e.g. for extensions to software already supplied

Stages in contract placement



Requirements document: sections

- introduction
- description of existing system and current environment
- future strategy or plans
- system requirements -
  - mandatory/desirable features
- deadlines
- additional information required from bidders

Requirements

- These will include
  - functions in software, with necessary inputs and outputs
  - standards to be adhered to
  - other applications with which software is to be compatible
  - quality requirements e.g. response times

Evaluation plan

- How are proposals to be evaluated?
- Methods could include:
  - reading proposals
  - interviews
  - demonstrations
  - site visits
  - practical tests

Evaluation plan -contd.

- Need to assess value for money (VFM) for each desirable feature
- VFM approach an improvement on previous emphasis on accepting lowest bid

Invitation to tender (ITT)

- Note that bidder is making an *offer* in response to ITT
- *acceptance* of offer creates a *contract*
- Customer may need further information
- Problem of different technical solutions to the same problem

Memoranda of agreement (MoA)

- Customer asks for technical proposals
- Technical proposals are examined and discussed
- Agreed technical solution in MoA
- Tenders are then requested from suppliers based in MoA
- Tenders judged on price
- Fee could be paid for technical proposals by customer

Contracts

- A project manager cannot be expected to be a legal expert – needs advice
- BUT must ensure contract reflect true requirements and expectations of supplier and client

Contract checklist

- Definitions – what words mean precisely e.g. ‘supplier’, ‘user’, ‘application’
- Form of agreement. For example, is this a contract for a sale or a lease, or a license to use a software application? Can the license be transferred?
- Goods and services to be supplied – this could include lengthy specifications
- Timetable of activities
- Payment arrangements – payments may be tied to completion of specific tasks

Contract checklist - continued

- Ownership of software
  - Can client sell software to others?
  - Can supplier sell software to others? Could specify that customer has ‘exclusive use’
  - Does supplier retain the copyright?
  - Where supplier retains source code, may be a problem if supplier goes out of business; to circumvent a copy of code could be deposited with an **escrow** service

Contract checklist - continued

- Environment – for example, where equipment is to be installed, who is responsible for various aspects of site preparation e.g. electricity supply?
- Customer commitments – for example providing access, supplying information
- Standards to be met

Contract management

Some terms of contract will relate to management of contract, for example,

- Progress reporting
- Decision points – could be linked to release of payments to the contractor
- Variations to the contract, i.e. how are changes to requirements dealt with?
- Acceptance criteria

Contract management

- Contracts should include agreement about how customer/supplier relationship is to be managed e.g.
  - *decision points* - could be linked to payment
  - *quality reviews*
  - *changes to requirements*

Managing people

- Managing people working as individuals and in groups
- People are an organisation's most important assets

Management activities

- Problem solving (using available people)
- Motivating (people who work on a project)
- Planning (what people are going to do)
- Estimating (how fast people will work)
- Controlling (people's activities)
- Organizing (the way in which people work)

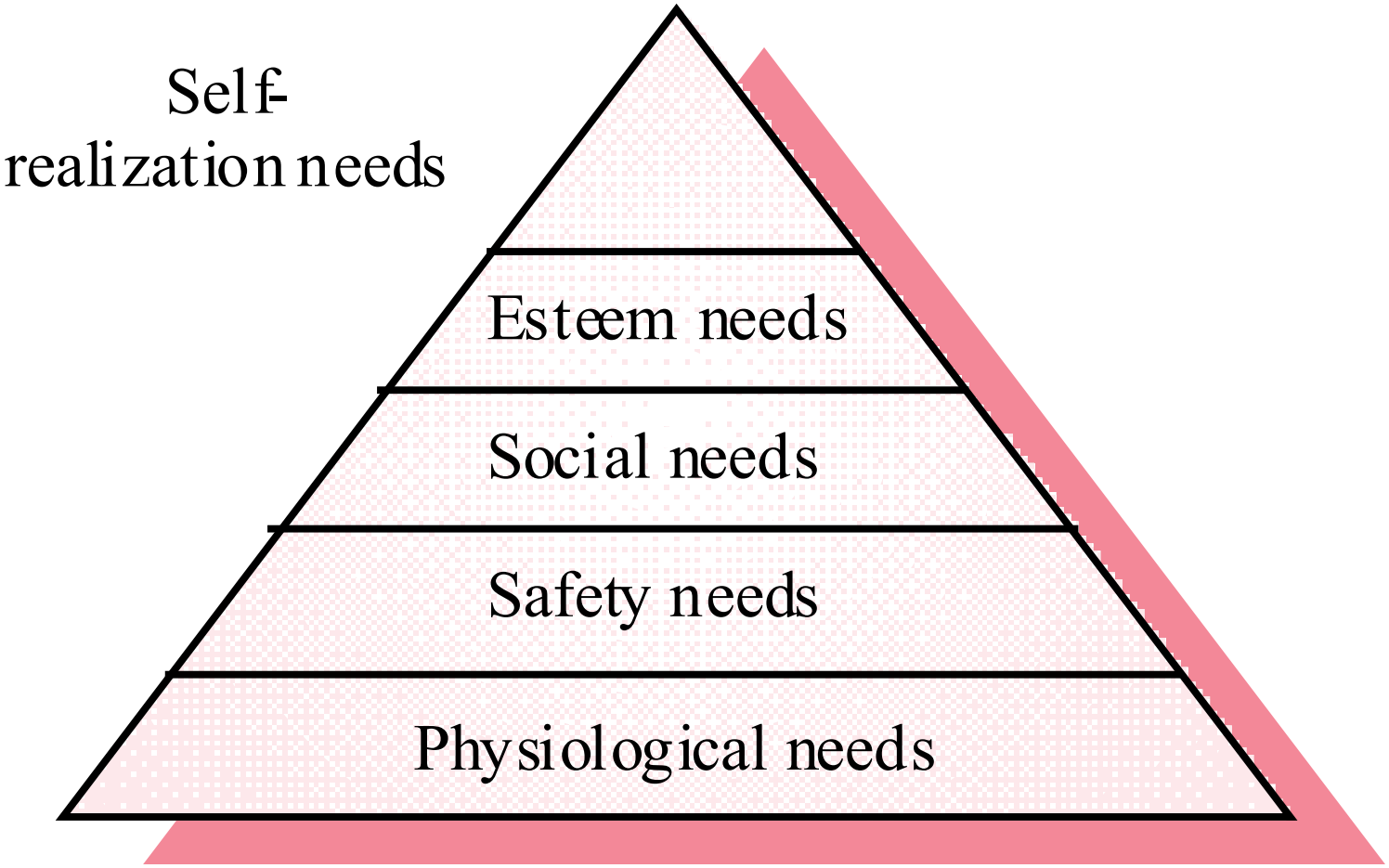
Problem solving

- Requires the integration of different types of knowledge (computer, task, domain, organisation)
- Development of a model of the solution and testing of this model against the problem
- Representation of this model in an appropriate notation or programming language

Motivation

- An important role of a manager is to motivate the people working on a project
- Motivation is a complex issue but it appears that their are different types of motivation based on
  - Basic needs (e.g. food, sleep, etc.)
  - Personal needs (e.g. respect, self-esteem)
  - Social needs (e.g. to be accepted as part of a group)

Human needs hierarchy



Personality types

- The needs hierarchy is almost certainly an over-simplification
- Motivation should also take into account different personality types:
  - Task-oriented
  - Self-oriented
  - Interaction-oriented

Personality types

- Task-oriented.
  - The motivation for doing the work is the work itself
- Self-oriented.
  - The work is a means to an end which is the achievement of individual goals - e.g. to get rich, to play tennis, to travel etc.
- Interaction-oriented
  - The principal motivation is the presence and actions of co-workers. People go to work because they like to go to work

Group working

- Most software engineering is a group activity
  - The development schedule for most non-trivial software projects is such that they cannot be completed by one person working alone
- Group interaction is a key determinant of group performance
- Flexibility in group composition is limited
  - Managers must do the best they can with available people

Group composition

- Group composed of members who share the same motivation can be problematic
  - Task-oriented - everyone wants to do their own thing
  - Self-oriented - everyone wants to be the boss
  - Interaction-oriented - too much chatting, not enough work
- An effective group has a balance of all types
- Can be difficult to achieve because most engineers are task-oriented
- Need for all members to be involved in decisions which affect the group

Group leadership

- Leadership depends on respect not title or status
- There should be both a technical and a managerial leader
- A career path based on technical competence should be supported

Group cohesiveness

- In a cohesive group, members consider the group to be more important than any individual in it
- Advantages of a cohesive group are:
  - Group quality standards can be developed
  - Group members work closely together so inhibitions caused by ignorance are reduced
  - Team members learn from each other and get to know each other's work
  - Egoless programming where members strive to improve each other's programs can be practised

Group communications

- Good communications are essential for effective group working
- Information must be exchanged on the status of work, design decisions and changes to previous decisions
- Good communications also strengthens group cohesion as it promotes understanding

Group communications

- Status of group members
  - Higher status members tend to dominate conversations
- Personalities in groups
  - Too many people of the same personality type can be a problem
- Sexual composition of group
  - Mixed-sex groups tend to communicate better
- Communication channels
  - Communications channelled though a central coordinator tend to be ineffective

Group organisation

- Software engineering group sizes should be relatively small (< 8 members)
- Break big projects down into multiple smaller projects
- Small teams may be organized in an informal, democratic way
- Chief programmer teams try to make the most effective use of skills and experience

Choosing and keeping people

- Choosing people to work on a project is a major managerial responsibility
- Appointment decisions are usually based on
  - information provided by the candidate (their resume)
  - information gained at an interview
  - recommendations from other people who know the candidate
- Some companies use psychological or aptitude tests
  - There is no agreement on whether or not these tests are actually useful

Factor	Explanation
Application domain experience	For a project to develop a successful system, the developers must understand the application domain.
Platform experience	May be significant if low-level programming is involved. Otherwise, not usually a critical attribute.
Programming language experience	Normally only significant for short duration projects where there is insufficient time to learn a new language.
Educational background	May provide an indicator of the basic fundamentals which the candidate should know and of their ability to learn. This factor becomes increasingly irrelevant as engineers gain experience across a range of projects.
Communication ability	Very important because of the need forproject staff to communicate orally and in writing with other engineers, managers and customers.
Adaptability	Adaptability may be judged by looking at the different types of experience which candidates have hadThis is an important attribute as it indicates an ability to learn.
Attitude	Project staff should have a positive attitude to their work andshould be willing to learn new skills. This is an important attribute but often very difficult to assess.
Personality	Again, an important attribute but difficult to assess. Candidates must be reasonably compatible with other team members. No particular type of personality is more or less suited to software engineering.

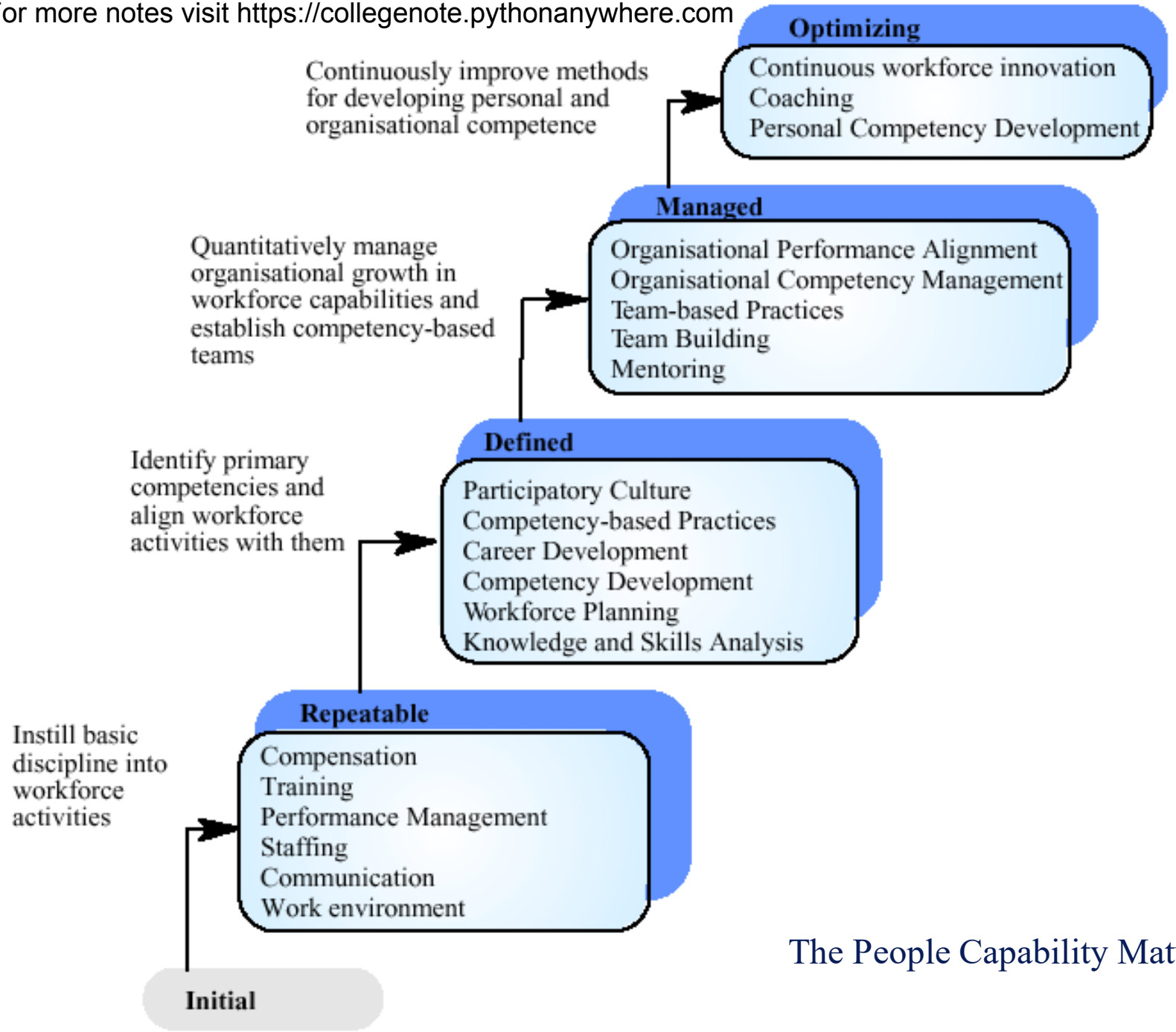
Staff selection factors

Working environments

- Physical workplace provision has an important effect on individual productivity and satisfaction
  - Comfort
  - Privacy
  - Facilities
- Health and safety considerations must be taken into account
  - Lighting
  - Heating
  - Furniture

The People Capability Maturity Model

- Five stage model
  - Initial. Ad-hoc people management
  - Repeatable. Policies developed for capability improvement
  - Defined. Standardized people management across the organization
  - Managed. Quantitative goals for people management in place
  - Optimising. Continuous focus on improving individual competence and workforce motivation



The People Capability Maturity Model