Lecture Objectives

- At the completion of this lecture you should:
  - be aware of the significance of maintenance activity in the work of an IS department
  - be aware of the types of maintenance activity that IS departments may carry out and the role of each type
  - be aware of the tasks involved in the maintenance of an information system

Transition to Maintenance

- “Once the product has passed its acceptance test, it is handed over to the client. Any changes after the client has accepted the product constitute maintenance.”
  - Schach, 1993

- Maintenance can be up to 75% of IS department budget

System Costs to IS Department

- 50% of D.P. budget consumed by maintenance and enhancements
- Testing consumes about 50% of systems development costs
- Need to develop systems that are more correct, cheap to operate, maintain and modify

What is Maintenance

- Maintenance is NOT just bug-fixing!
- 4 types of maintenance
  - Corrective maintenance
  - Adaptive maintenance
  - Perfective maintenance
  - Preventative maintenance

Corrective Maintenance

- Corrects analysis, design and implementation errors
  - most corrective problems arise soon after installation or after major system changes
  - should have been isolated and corrected during development
  - professional practice during development should minimise the need (but will not remove it completely)
  - adds little or no value - focus on removing defects rather than adding anything new
  - accounts for up to 75% of all maintenance activity
## Maintenance

### Corrective Maintenance
- Can be the most expensive kind of maintenance
- Costs of functions not working correctly
- Requires immediate attention
- Typically urgent, interfere with normal operations
- Needs skilled maintenance staff to ensure rapid diagnosis of errors and their correction
  - Must have or quickly develop high level of familiarity with the system
  - Software tools for diagnosis

### Adaptive Maintenance
- To satisfy changes in the environment, changing business needs or new user requirements
  - Changes in tax laws, takeovers and mergers, new OS, etc.
  - New type of report, new class of customer etc.
  - Less urgent - changes occur over time
- Adaptive maintenance is inevitable, does add value
- Maintenance staff need strong analysis and design skills as well as programming skills
  - Changes often require a complete SDLC
  - Also need good understanding of the system

### Preventative Maintenance
- Pay now or pay more later
  - Defects or potential problems found and corrected before they cause any damage
  - Reduce chance of future system failure
  - Eg expand number of records beyond needs, standardise formats across platforms
- A natural by-product of maintenance work - identify and fix any potential problems noted while fixing other errors

### Preventative Maintenance
- Ideally have periodic (monthly / half-yearly / annual) reviews of system to uncover and anticipate problems - lower priority
- Often ignored because
  - Too few resources
  - Fear of recrimination - why wasn’t it done properly the first time?
  - No way of accounting for the cost of the effort

### Perfective Maintenance
- To enhance performance, maintainability, usability
  - Adds desired features rather than required
  - Better run times, faster transaction processing
- To meet user requirements not previously recognised or given high priority
  - Missed in development or not known about
  - Considered unimportant

### Perfective Maintenance
- Legacy systems (old systems running for at least 10 years) are likely candidates for perfective maintenance
- May involve technical systems specialists as well as general maintenance staff
  - Network specialist to change network design for improved performance

---

version: Jan 97  updated by H. Smith
Topic: Maintenance

Distribution of Maintenance Effort

- Corrective
- Adaptive
- Perfective
- Preventive

Costs of Maintenance

- There are many factors or cost elements affecting the 'maintainability' of a system
- Maintainability
  - the ease with which software can be understood, corrected, adapted and enhanced
  - Low maintainability results in uncontrollable maintenance expenses

Cost Elements of Maintenance

- The following factors affect 'maintainability'
  - Defects
  - Customers
  - Documentation
  - Personnel
  - Tools
  - Software structure
- Defects, customers and documentation have a significant effect on maintainability

Cost Elements of Maintenance

- Defects
  - the number of latent or unknown errors existing after system installation
  - influences most maintenance costs, drives all other cost factors
  - few errors --> low maintenance costs
- Customers
  - the number of customers/users of system
  - more customers, more maintenance effort/cost
  - greater need for high maintainability

Cost Elements of Maintenance

- Documentation
  - quality of system documentation
  - exponential effect on maintenance costs
- Personnel
  - quality of maintenance personnel
  - highly skilled programmers, typically not original programmers, to quickly understand and carefully change system
  - separate from development? in-house? dedicated end-user support?

Cost Elements of Maintenance

- Tools
  - appropriate automated development tools
  - programming tools, code generators, debuggers, hardware, CASE, diagnostics, etc
  - reverse engineering for no documentation
- Software structure
  - quality of software structure and maintainability
  - formalisation of code, comments, versioning
  - structure charts, OO

version: Jan 97 updated by H. Smith
Topic: Maintenance

### Measuring Maintenance Effectiveness
- There is a need to measure maintenance
- Understand quality of development/maintenance effort
- We measure the following factors
  - Number of failures
  - Time between each failure
  - Type of failure
- Mean Time Between Failures (MTBF)
  - Calculated using number of failures and time between each failure, widely used measure of quality

### Maintenance Life Cycle
- Software Maintenance Life Cycle (SMLC)
  - Receive a Maintenance Request
  - Transform the Maintenance Request to a Change (analysis)
  - Specify the Change (design)
  - Develop the Change (code)

### Maintenance Life Cycle
- Software Maintenance Life Cycle (SMLC)
  - Test the Change
  - Train users and run an acceptance test
  - Convert and release to operations
  - Update the documentation
  - Conduct a Post-Maintenance Review

Chapin, 1988

### Change Management Systems
- Overall goal is to manage change effectively
- Organisations implement change management systems in an attempt to reduce the confusion and complexity of developing and maintaining systems

### Change Management Systems
- The aims of change management systems are
  - Restrict access to production source and object code
  - Reduce errors being introduced into production
  - Single version of source and object code in production
  - Improve quality and reliability of software
  - Increase security and control
  - Increase software productivity

### Maintenance
- It is important to realise that adaptive, perfective, and preventive maintenance can lead to corrective maintenance if not carefully designed and implemented
- Occasionally system failure is inevitable!

version: Jan 97 updated by H. Smith
What maintenance is needed here?

- In March, 1999, a man living in Kandos (near Mudgee in NSW) received a bill for his as yet unused gas line stating that he owed $0.00. He ignored it and threw it away.
- In April he received another bill and threw that one away too.
- The following month the gas company sent him a very nasty note stating they were going to cancel his gas line if he didn't send them $0.00 by return mail. He called them, talked to them, and they said it was a computer error and they would take care of it.
- He called the gas company who apologised for the computer error once again and said that they would take care of it. The next day he got a bill for $0.00 stating that payment was now overdue. Assuming that having spoken to them the previous day the latest bill was yet another mistake, so he ignored it, trusting that the company would be as good as their word and sort the problem out.
- The next month he got a bill for $0.00. This bill also stated that he had 10 days to pay his account or the company would have to take steps to recover the debt.
- Finally, giving in, he thought he would beat the company at their own game and mailed them a cheque for $0.00. The computer duly processed his account and returned a statement to the effect that he now owed the gas company nothing at all.
- A week later, the manager of the Mudgee branch of the Westpac Banking Corporation called our hapless friend and asked him what he was doing writing cheques for $0.00.
- After a lengthy explanation the bank manager replied that the $0.00 cheque had caused their cheque processing software to fail. The bank could therefore not process ANY cheques they had received from ANY of their customers that day because the cheque for $0.00 had caused the computer to crash.

What maintenance is needed here?

- The matter was heard in the Magistrate's Court in Mudgee and the outcome was this:
  1. The gas company was ordered to:   
     1.1) Immediately rectify their computerised accounts system or show cause, within 10 days, why the matter should not be referred to a higher court for consideration under company Law.
     1.2) Pay the bank dishonour fees incurred by the man.
     1.3) Pay the bank dishonour fees incurred by all the Westpac clients whose cheques had been bounced on the day our friend's had been.
     1.4) Pay the claimant's court costs; and
     1.5) Pay the claimant a total of $1500 per month for the 5-month period March to July inclusive as compensation for the aggravation they had caused their client to suffer.

References

Chapter 21

Chapter 17

version: Jan 97  updated by H. Smith