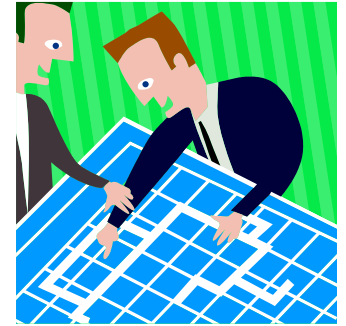




INFO5990 Professional Practice in IT

Lecture 13A

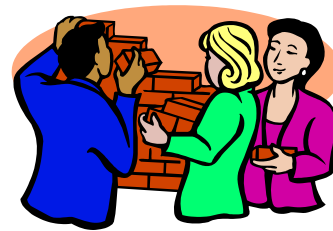


Where have we been?



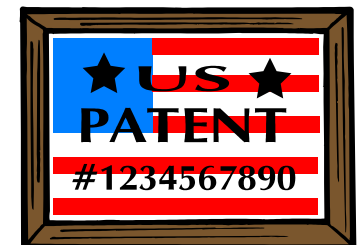
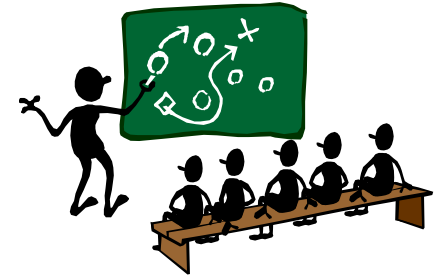
Learning outcomes (1) Useful Skills

1. Written communication
2. Oral presentation
3. Testing
4. Problem solving
5. Project management
6. Working in team



Learning outcomes (2) Professional Attitudes

1. Being an effective team player
2. Being an effective leader
3. Acting 'ethically' at all times
4. Dealing with privacy and security
5. Respecting intellectual property



Learning outcomes (3) Valuable Experiences



1. Formal writing
2. Giving an oral presentation
3. Working as part of a team
4. Forming a network of friends

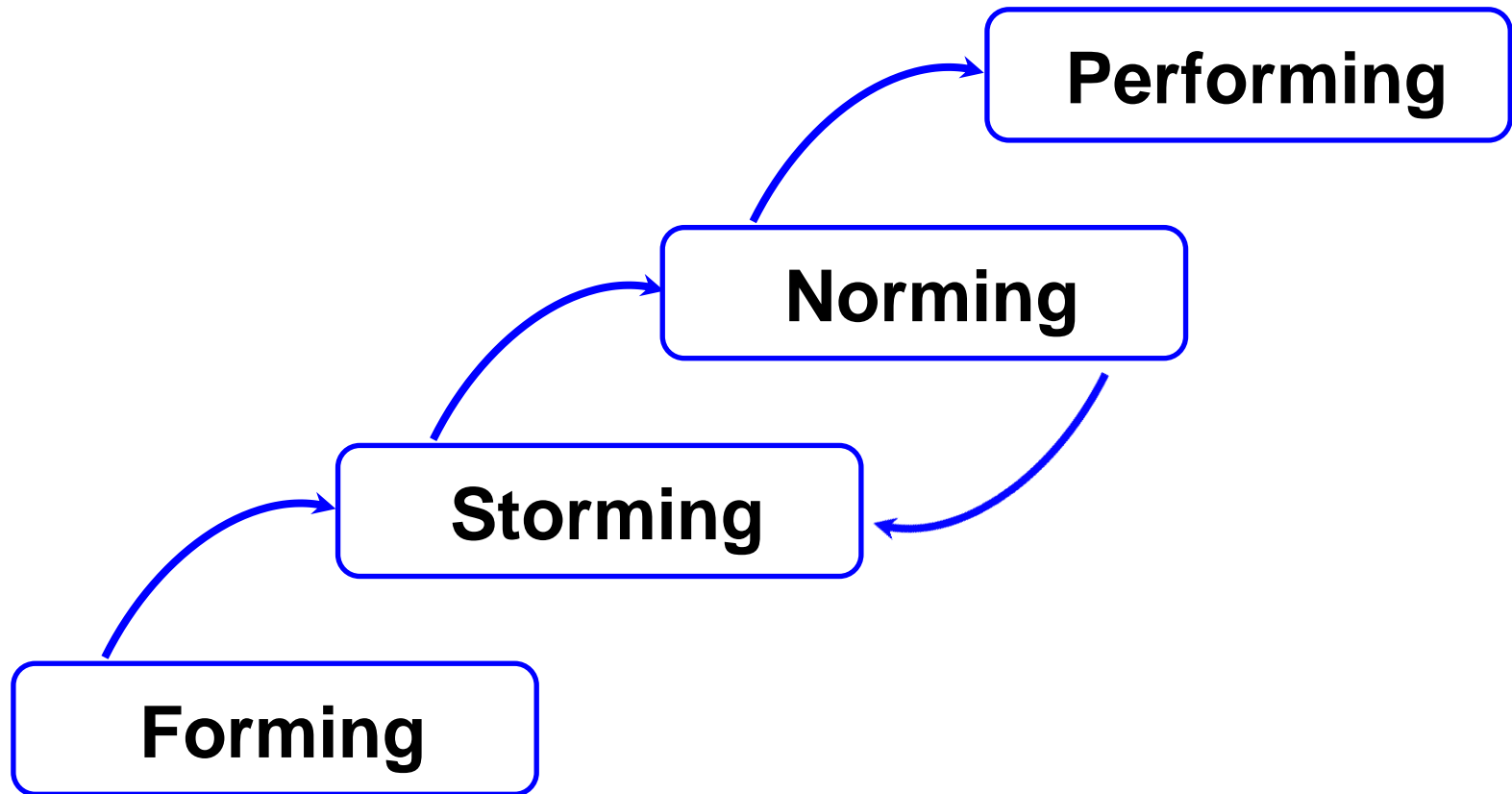


Stages of Group Development

- Devised by Bruce W Tuckman in 1965
- The model was further refined in 1977 when a new stage was added.



Stages of team development



Did your team display any of the following characteristics?

- High level of negativity and passivity
- Quick (and 'dirty') problem solving; lack of clarity about what problem is
- Lot of win-lose situations among members
- Strangled information flow; dominance by one or two members; power plays
- Mistaking silence for support

Anat Lechner, (2003) Stern School of Business

http://sydney.edu.au/engineering/it/~isys3207/readingsontteams/groupsb2351_lechner.pdf

Nurturing and Retaining IT Talent

Top 3 attraction and retention strategies:

- Differentiate compensation based on performance
- Provide access to coaching and mentoring
- Offer job and career flexibility

The Deloitte and CIO survey 2009:

<http://www.cio.com/documents/whitepapers/mindthetalentgap.pdf>

Change management

The number one obstacle to success
for major change projects is:

**“employee resistance and the
ineffective management of the
people side of change”**



The ADKAR[®] Change Model

Five building blocks for successful change

Awareness

- Of the need to change
- Of the nature of the change

Desire

- To support the change
- To participate and engage

Knowledge

- What to change
- How to implement new skills and behaviors

Ability

- To implement the change
- To demonstrate performance

Reinforcement

- To sustain the change
- To build a culture and competence around change



Example: Introducing 'flexi-time'

Phase

What change manager needs to do

- **A**wareness
 - ❑ Explain why the change, what will be different
- **D**esire
 - ❑ Appreciate the advantages, want to help colleagues participate
- **K**nowledge
 - ❑ How to go about it, what steps you need to take, any limitations
- **A**bility
 - ❑ Ensure that procedures are in place, documents as simple as possible
- **R**einforcement
 - ❑ Acknowledge those taking advantage, develop organisation culture

Information System Audit: 3 key areas of concern

Availability	<ul style="list-style-type: none">• Will systems be available for the business at all times when required?• Are systems well protected against all types of losses and disasters?
Confidentiality	<ul style="list-style-type: none">• Will information in the system be disclosed only to those who need it?
Integrity	<ul style="list-style-type: none">• Will the information provided in the system always be accurate, reliable and timely?• What ensures that no unauthorized modifications?

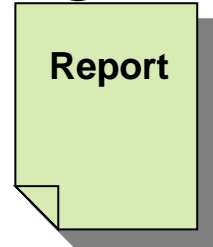
S.Anantha Sayana : <http://www.isaca.org/Journal/Past-Issues/2002/Volume-1/Pages/The-IS-Audit-Process.aspx>

Risk Based Approach

- A useful approach where cost prevents auditing all aspects of a system
- Order priorities on basis of risk
- Risk can affect various systems differently
 - Non availability of a system
 - Compare Restaurant booking system with Company Blog
 - Unauthorised access to a system
 - Compare Online Banking with Hair Salon booking system

Characteristics of professional writing

- Clarity
 - Clear structure. Logical arrangement
 - Simple but complete explanations
- Precision
 - No ambiguity or confusion
 - Uses words with precise meaning
- Objectivity
 - Statements supported by evidence
 - Avoids exaggeration or emotive statements
- Brevity
 - Effective and efficient
 - Avoids being 'longwinded', stating the obvious



$$E = mc^2$$



Aspects of an oral presentation

1. The Message
2. Structure
3. Timing
4. Physical factors
5. Personal factors
6. Visuals - Powerpoint



Benefits of using styles and templates

- Consistency
- Flexibility
- Productivity
- Templates define the ‘typography’ for a your document
- You can use the same template over and over again

Four most common reasons for project failure

- Incomplete project **requirements** (10%)
- Unrealistic **schedules** (13%)
- Insufficient resource **planning** (18%)
- Poor **communications** (28%)

These are four areas that are central to the practice of

“PROJECT MANAGEMENT”

What is a project?

- “A project is a temporary endeavour undertaken to create a unique product, service, or result”.
 - *temporary* = a definite beginning and end
 - end = objectives have been achieved, or,
 - the project is terminated because its objectives will not or cannot be met, or when the need for the project no longer exists.
 - *temporary* does not necessarily mean short in duration.

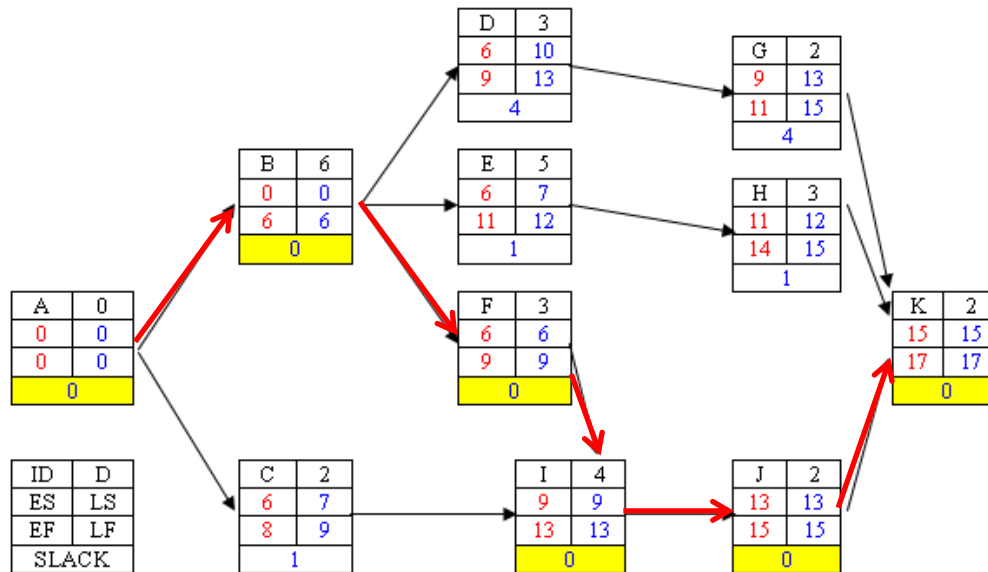
“A Guide to the Project Management Body of Knowledge”, PMI Standards Committee

Work breakdown structure

- Lists all of the work that needs to be done
- Helps define the scope of the project
(can also state what is NOT to going be done)
- Basic input to all aspects of the planning and scheduling process
- Lowest level of subdivided work should be
 - small enough to permit adequate control and visibility
 - without creating an unwieldy administrative burden. (Charles C. Martin)

Network diagrams

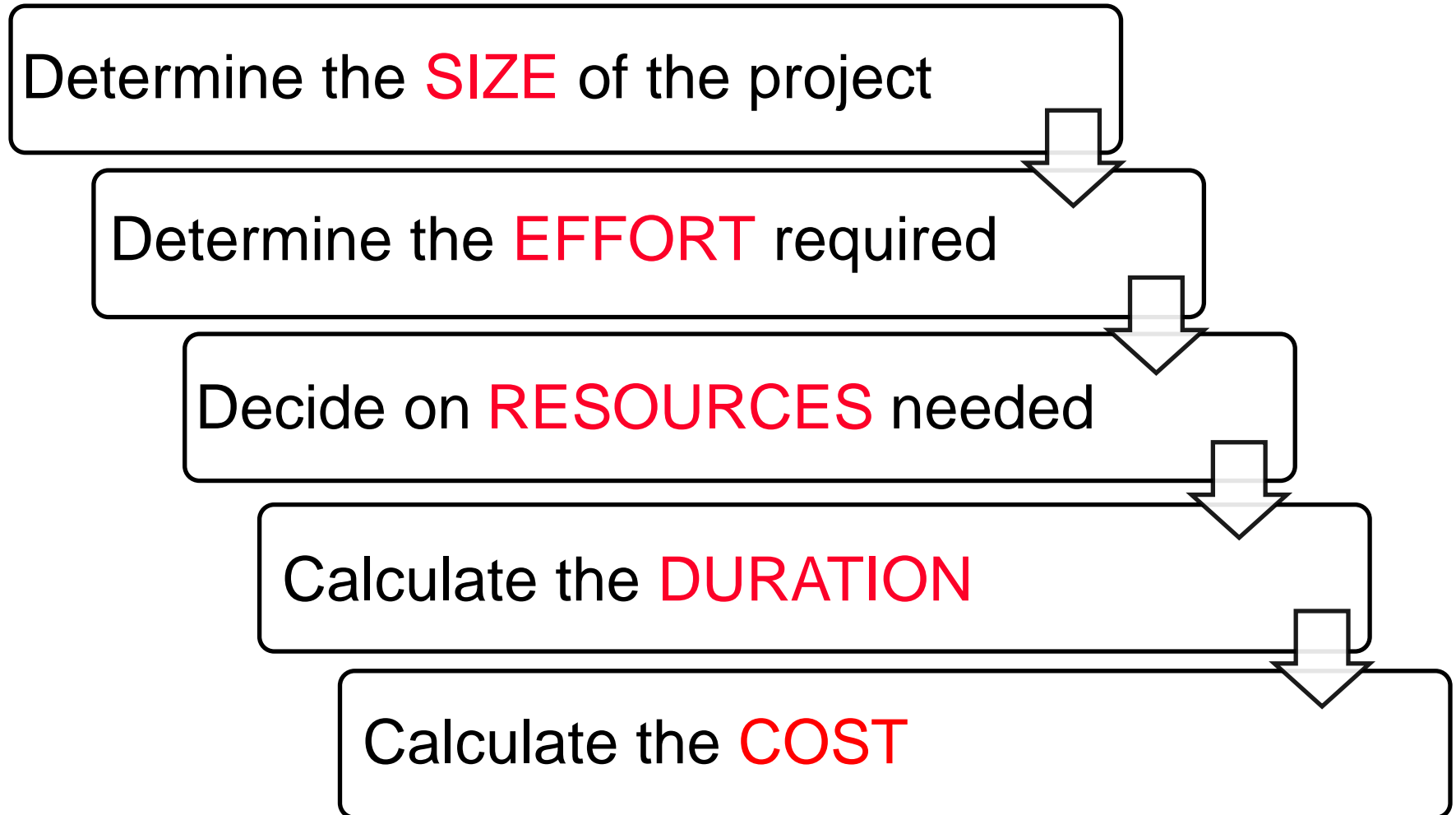
- Critical path method (CPM)
 - The longest path through the network
 - Passes through all nodes with zero slack
- Program Evaluation and Review Technique (PERT)
 - Makes use of three estimates: best, optimistic, pessimistic



Monitoring progress

- Limits to reducing the duration
 - Available resources: e.g. programmers or engineers with the necessary skills
 - Budget: the cost increases as more resources used
 - Communication: becomes more of a problem
 - Management of the project: becomes more difficult as size of team increases

Flow chart of project estimation



Six approaches to project estimation

1. Expert judgement
2. Sum of the parts
3. Estimation by analogy
4. Component matrix
5. Algorithmic cost models
6. Function point analysis

What is an empirical model?

Trend Line EFFORT v SIZE



Choosing estimation method

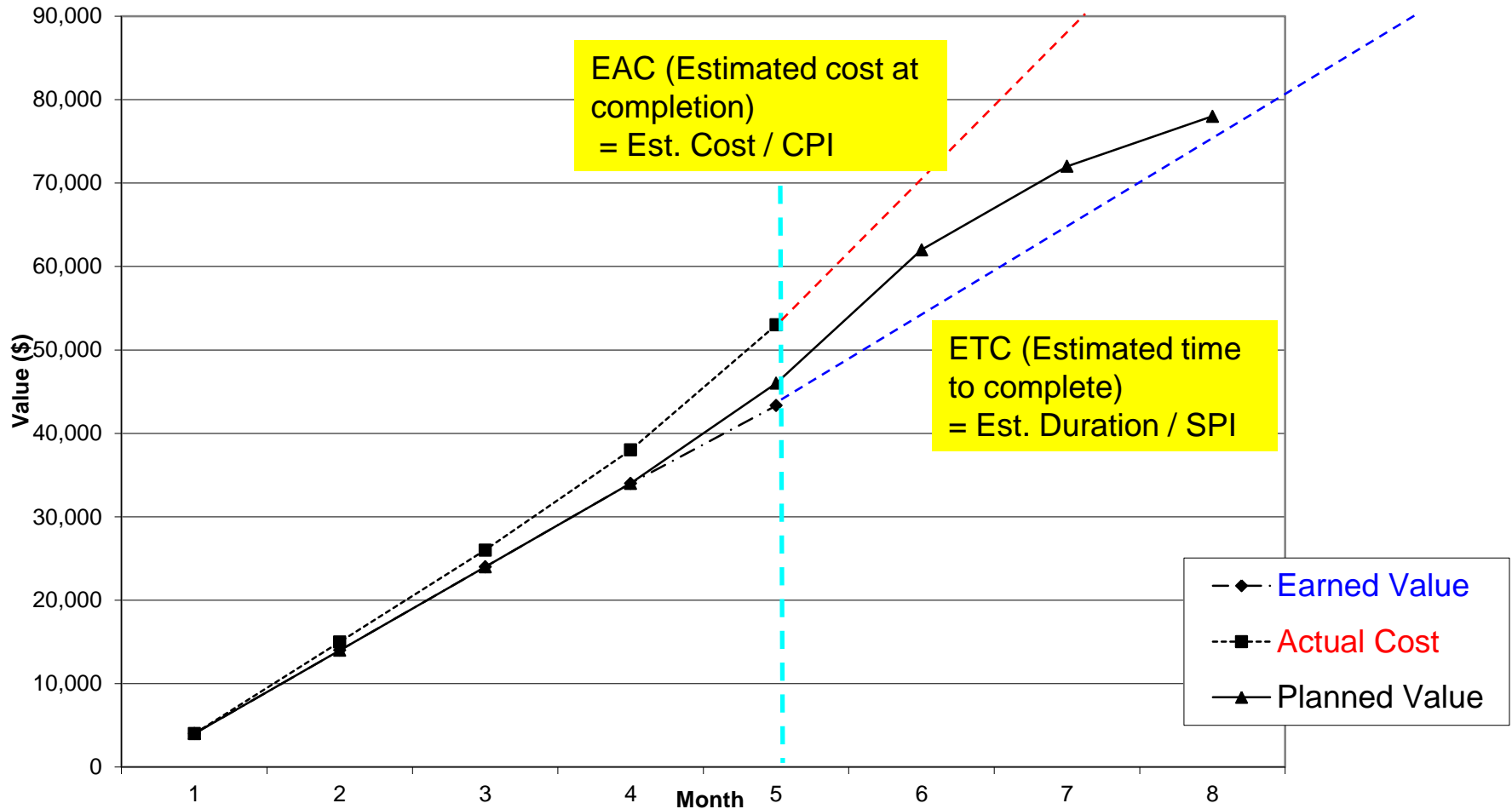
- Which is easiest to apply?
- Which can be applied earliest in the system development life cycle (SDLC)?
- What assumptions does each make?
- Do I have enough historic data?
- Will I need to re-calibrate for tool, developer experience, environment, etc.
- How many times will I use this method?
Can I improve my estimates over time?

Earned value and key performance indicators

- Cost Performance Index
 - $CPI = \text{Earned value} / \text{Actual cost}$
 - 1.00 or positive is good (on track)
 - Less than 1 is 'bad' (budget over spent)
- Schedule Performance Index
 - $SPI = \text{Earned value} / \text{Planned value}$
 - 1.00 or positive is good (on track)
 - Less than 1 is 'bad' (budget over spent)

How will we end up if work continues at the same rate?

Earned Value Chart



What NOT to do if the project is 'slipping'

- Ignore it
- Try to hide it
- Keep on keeping on
- Rob Peter to pay Paul
- Try to play 'catch up'



Responses to Risk

1. Avoidance
2. Mitigation
3. Transfer
4. Sharing
5. Acceptance

Three essential software attributes

1. Correct

- behaves according to the functional requirements
- produces the right result for any given set of inputs

2. Reliable

- behaves as expected on every occasion
- over any period of time

3. Robust

- behaves in a predictable and controllable fashion even if the input not valid.
- a program may be correct but not robust
 - e.g. division by zero or non-numeric input.

Terms used in connection with software testing

- *A defect*

- any error or mistake, which may cause the program to not perform according to specifications.
- may be the result of programmer error, or of incorrect or incomplete specifications
- may not always cause the program to fail.

- *A failure*

- occurs when the software does not behave as expected.

Test cases

- To define a 'test case' you must say
 1. What function is being tested [Function]
 2. The value(s) of the test input [Input]
 3. The result or output expected [Result]
 - output value
 - error message
 - further process
- One or more test cases are required for each function or rule specified by the client

Equivalence class example

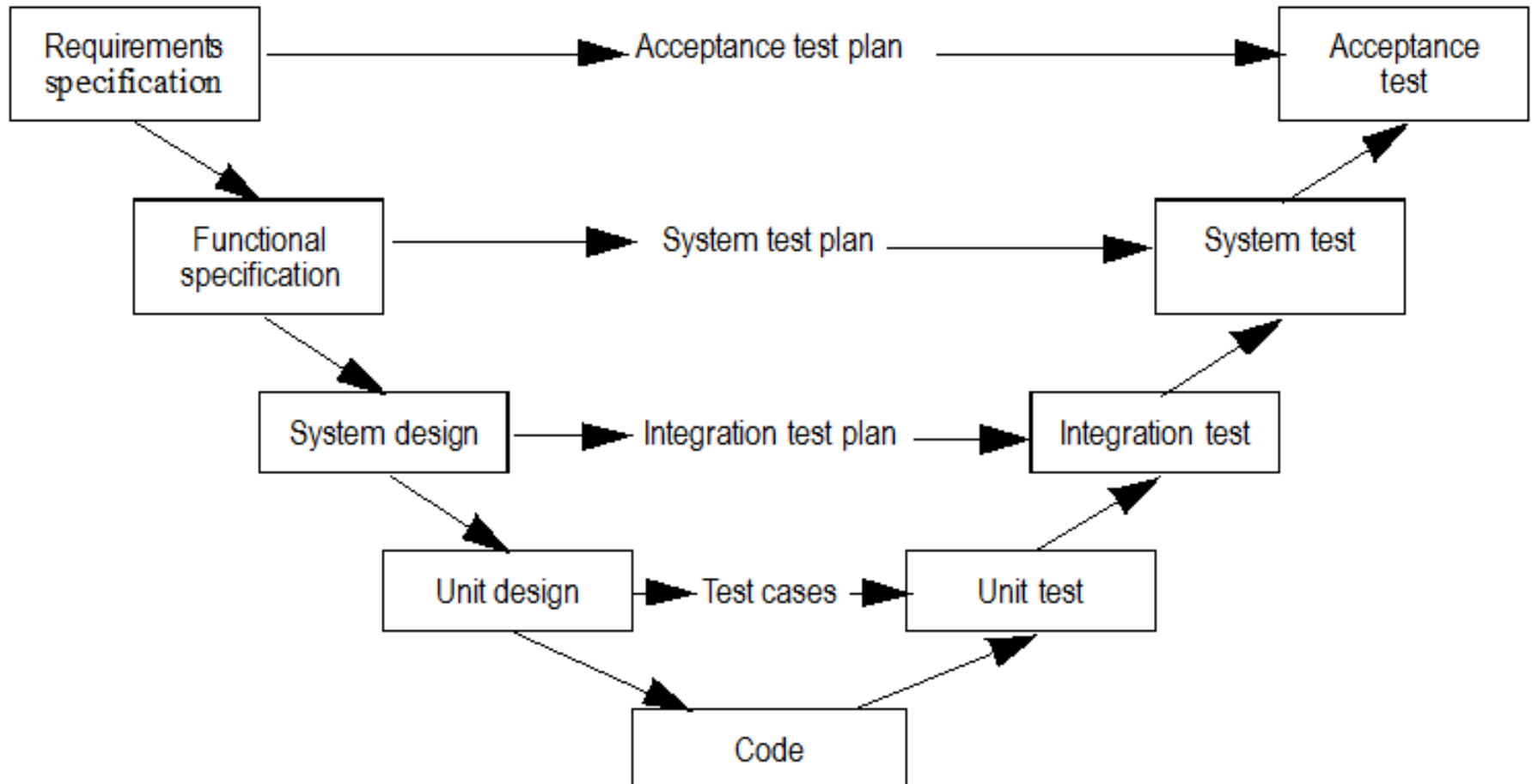
- Logic to be tested “Quantity discount example”
 - If quantity > 5
Then discount = 5%
Else discount = 0%
- Equivalence classes
 - {1,2,3,4}
 - {6,7,8,...}
- Boundary value
 - {5}
- Special case
 - {0}

The V-Model

Verify against the specification.

The V-Model implies a policy of continuous testing

Validate against client requirements



Difficulty of testing usability

- To test the **usability** of new products must have skilled, highly-experienced users

So, have to **train** subjects to high levels of skill before can even begin the experiment...

- “**Intuitive**” and “**easy to learn**” may not be synonymous when evaluating software, but often they are taken to be equivalent.

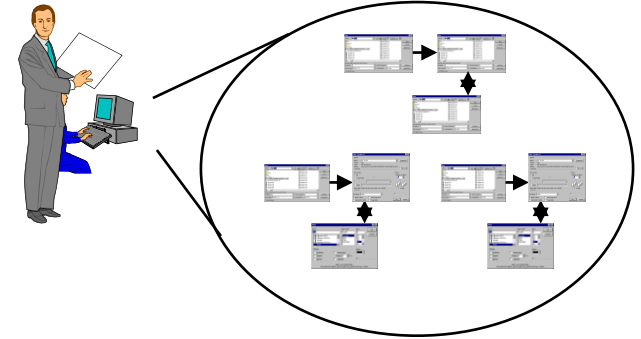
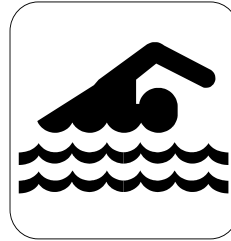
Larry Constantine, “*Persistent Usability*”, 1994, *OzCHI Proceedings, Australian Centre for Human-Computer Interaction*

Shneiderman's usability criteria*

1. Time taken to learn basic/advanced skills
2. Speed of performance of *skilled* users
3. Retention of syntactic knowledge over time
4. Error rates and ease of correction
5. Subjective satisfaction

* *Ben Shneiderman, "Designing the User Interface", Addison-Wesley 1987*

Implementation Testing



Performance Test

- To test system performance under maximum expected load.
- Simulates key processes under maximum load.

Soak Test and Stress Test

- To ensure that system is stable over extended period.
- Load increased until system fails. Checks effects of over-load

Acceptance Test

- Compares system functionality against agreed-on user requirements
- Carried out by client using scenarios, supervised by developer

A Bitter experience

‘Nectar’ card launch fiasco

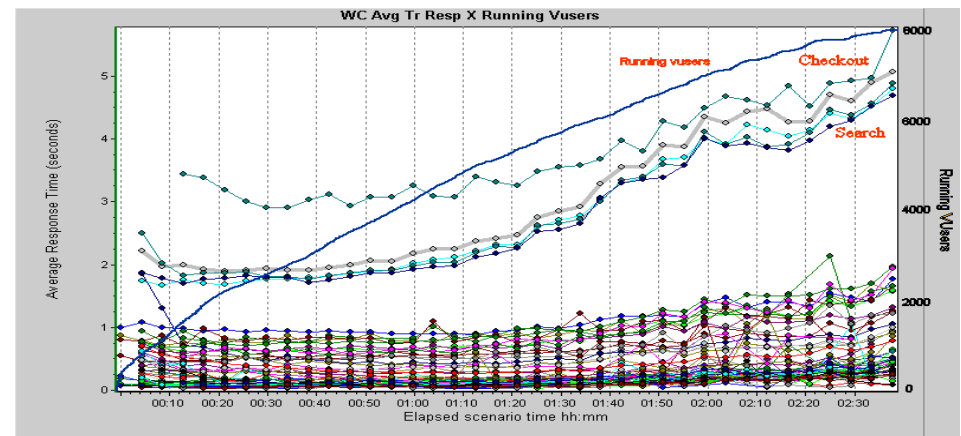


System testing for

- Total budget for testing, £1 million
- Testing to be carried out off line (so as not to interfere with live system)
 - Capacity model: to simulate user load two years into future
 - Usage model: typical mix of tasks
 - Test database: full-sized database, since size affects performance

Testing approach

- Usage model
- Capacity model
- Use *Mercury* 'LoadRunner' to emulate thousands of concurrent users to simulate real-life user loads
- Results can be analysed to measure performance, identify bottlenecks, etc.
- Load test, stress test, soak test



The TJX case: 17 Jan, 2007



- TJX retailers
 - 2100 stores in US, 300 in Canada
 - \$16 billion annual revenue
- “The worst retail data breach ever?”
 - 46 million customers affected
- Details
 - What happened?
 - How did it happen?
 - What was the result?
 - What lessons?



Australian Computer Crime & Security Survey 2006

- Sample: 2,024 survey forms distributed
- 389 responses received from wide range of organisations (17%)
- 22% of respondents (86) suffered 1 - 5 computer security incidents in 2006
 - Compared with previous years
 - 49% in 2004
 - 67% in 2002

Reporting incidents to law enforcement authorities

- 22% reported the incident
 - 69% of those affected chose not to report
- Reason given for not reporting
 - Not considered serious enough (76%)
 - Didn't think perpetrators would be caught (57%)
 - Didn't think authorities were competent (55%)
 - Wanted to avoid negative publicity (46%)

Five categories of security threats

1. Unintentional acts
 - Human error, carelessness, ignorance
2. Natural disasters
 - Power outage, fire, flood, earthquake
3. Technical failures
 - Hardware failure, software failure
4. Management failures
 - Ineffective procedures and controls
5. Deliberate acts
 - Vandalism and malicious damage
 - Malware: viruses, worms and spyware
 - Phishing

Ethical theories

- Consequentialism
 - The evaluation of an action depends on the non-moral consequences that the action brings about.
- Utilitarianism
 - The moral standard should be the promotion of the best long-term interests of everyone concerned, i.e. positive consequences for everyone
- Deontology

(from Greek δέον, deon, “obligation, duty”)

 - ‘Goodness’ or ‘rightness’ is determined by examining the acts themselves, rather than consequences, or even the intentions of the person doing the act.

The burden of being a “professional”

- Responsibility: A moral concept
 - You accept the consequences and obligations of your decisions and actions, as a member of the human race
- Accountability: An organisational concept
 - You need to be able to justify your decisions on rational grounds and be prepared to be held responsible for such decisions
- Liability: A legal concept
 - Existing laws give any individuals affected, including those downstream, the right to recover costs resulting from your actions

What is a Professional Body?

“a group of people in a learned occupation who are entrusted with maintaining control or oversight of the legitimate practice of the occupation”

(www.qualityresearchinternational.com)



Australian Computer Society Code of Ethics



(7 Oct, 2009)

To uphold and advance the honour, dignity and effectiveness of the profession of information technology and in keeping with high standards of competence and ethical conduct, a member must:

- (a) be honest, forthright and impartial
- (b) loyally serve the community, and
- (c) strive to increase the competence and prestige of the profession
- (d) use special knowledge and skill for the advancement of human welfare.

http://www.acs.org.au/attachments/Code_of_Ethics.pdf

Ethics case studies

1. The Flaw in Intel Pentium chip (1994)
2. The Bay Area Rapid Transport (BART) Case (1971)
3. Low frequency electromagnetic fields (1994)
4. Paradyne computers (1980)

What is Intellectual Property (IP) ?

- Represents the property of your mind or intellect.
- It can be worth money and may be sold on to other parties to utilise
- It may give you the 'edge' which will make your company successful
- It may be stolen and/or used without permission

Patents

- Can only apply to **technology** , i.e. something that is a product, a composition or a process.
 - Must be **novel**, i.e. different from anything that has gone before.
 - Must be **useful**, i.e. have the potential for commercial return
 - Must be **inventive**, i.e. the result of some ingenuity on your part, not just a solution to a problem that would have been obvious to anyone.

Patenting Software

- A claim containing a mathematical formula may be patentable
 - “if it implements or applies the formula in a structure or process which, when considered as a whole, is performing a function which the patent laws were designed to protect”
- You can patent computer software, if it can be shown to have practical use that results in commercial returns
- In practice most software developers prefer to rely on copyright to protect their programs

Copyright in Australia

- Free, automatic, does not have to be registered
- Life of the copyright
 - pre 1955: 50 years after the author's death
 - post 1955: 70 years after the author's death
- It gives the owner rights to license others to copy, perform or broadcast the work

10 Copyright myths

Art Majlessi, 2004,

<http://www.legalmetro.com/library/copyright-law-explained.html>

1. "If it doesn't have a copyright notice, it's not copyrighted."
 - False. According to the Berne copyright convention, anything created privately after April 1, 1989 is copyright.
 - It does not need the © symbol or anything else.
2. "If I don't charge for it, it's not a violation."
 - False, but it may matter if the court decides to award monetary damages.
3. "If it's posted to the internet it's in the public domain."
 - False. Copyright law still applies and the copyright still belongs to the author.
4. etc.

Problem Solving

- A problem occurs when a system
 - does not meet its established goals
 - does not yield the predicted results, or
 - does not work as planned
- To solve the problem is to reduce the difference between the desired outcome and the actual outcome
- Problem solving also includes the identification of new opportunities

Benefits of Mathematical Modeling

- Models are easy to manipulate
- Facilitates compression of time
- Permits evaluation of many alternatives
- Cheaper than building scale models
- No cost of making mistakes in experiments
- Can build in risk and uncertainty (stochastic models)

Optimization models: e.g. Linear program

$$\text{Maximize } 7T + 5C \quad (\text{profit})$$

Where T = Number of Tractors to make

C = Number of Cars to make

Subject to the constraints:

$$3T + 4C \leq 2400 \quad (\text{limited finishing hours})$$

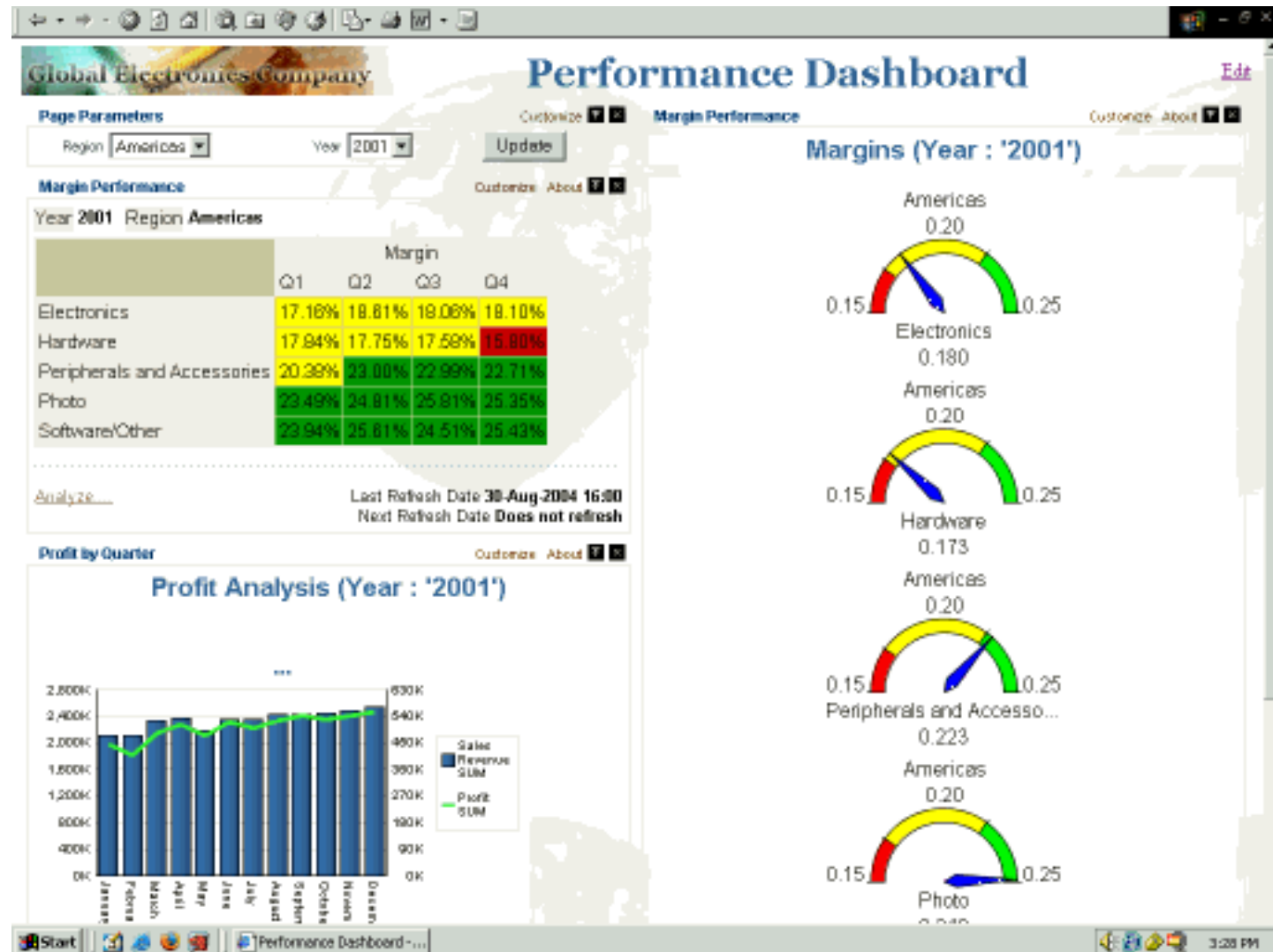
$$2T + 1C \leq 1000 \quad (\text{limited testing hours})$$

$$0 \leq C \leq 450 \quad (\text{max. number of Cars})$$

$$T \geq 100 \quad (\text{min. number of Tractors})$$

Maximum profit is \$4,040 when
number of tractors, $T = 320$ and
number of cars, $C = 360$

Monitoring performance in 2012: the dashboard



ETL tools

Extract-Transform-Load

- Extract data from multiple diverse data sources including those outside the organisation
- Transform data to fit operational needs, including 'cleansing' (quality)
- Load data into target database, data mart or data warehouse
- ETL 'World Record':
5.4 TB data loaded in Under 1 Hour (Syncsort)

Dimensions, hierarchies & measures

- Dimensions:

- aspect of business: state, store, product, time-period, actual and budget expenses

- Hierarchies:

- each level can be a 'child' of the previous parent level, e.g. Year, Quarter, Month, Day, time of day

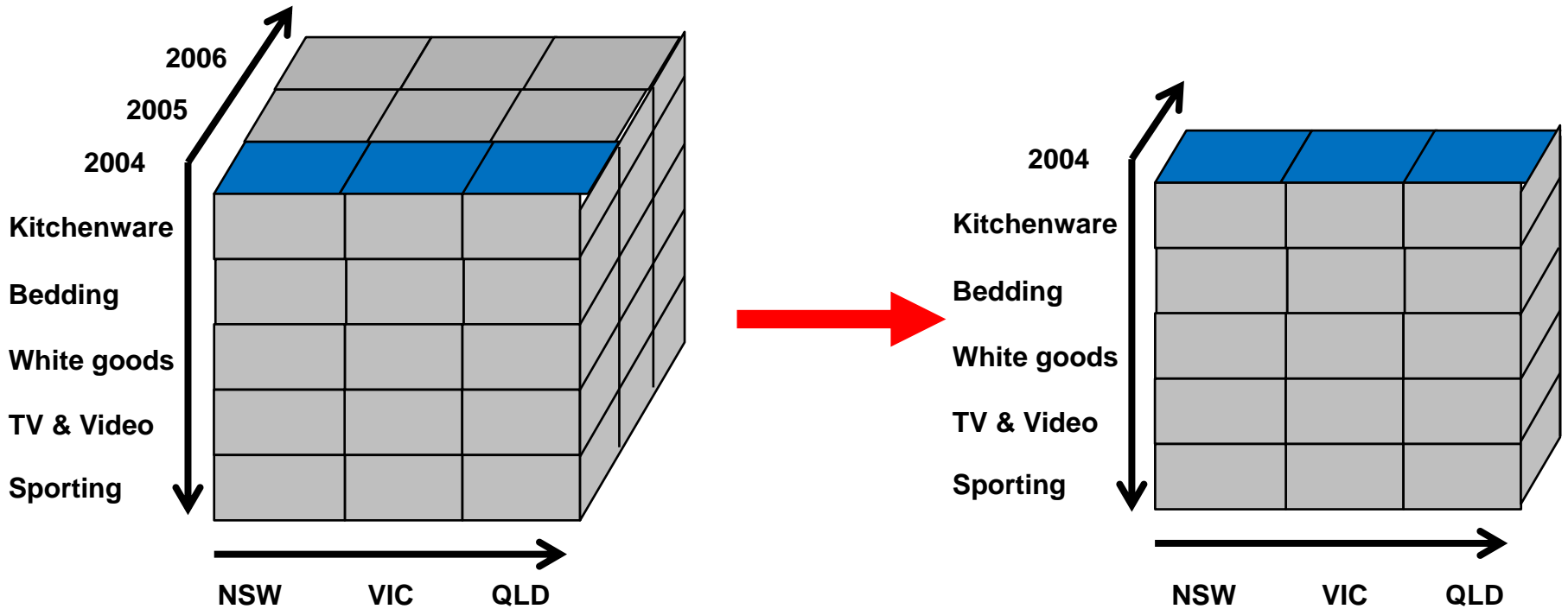
- Measures:

- Each cell of cube holds a number, some fact about the business, e.g. sales, profits, expenses, budget, forecast

- Grain:

- A question of how finely grained to store data

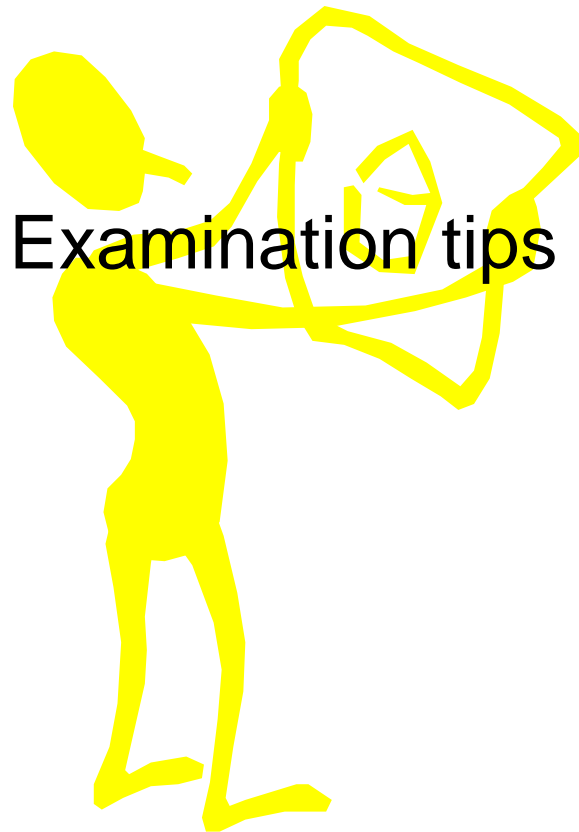
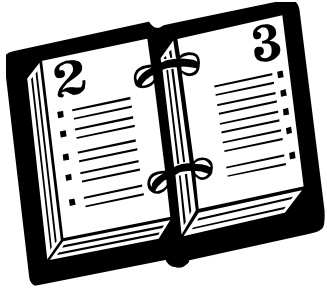
OLAP operations: Slicing



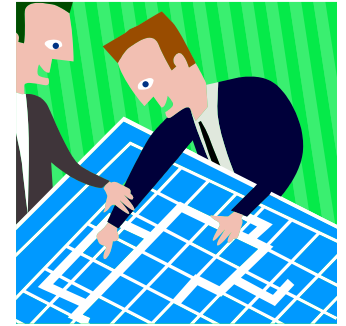


INFO5990 Professional Practice in IT

Lecture 13B



Examination tips



Examination

INFO5990 Professional Practice in IT

Time:

Tuesday 12th November 2013

06:00 PM (2 Hours)

Location:

XXXXXXXXXXXXXXXXXXXX

Simple non-programmable calculators permitted

Front page

Front page layout template showing the University of Sydney logo and various form fields.

THE UNIVERSITY OF SYDNEY

Form fields include:

- Header section with dotted lines for text entry.
- A large rectangular box for a title or main heading.
- A section with multiple lines of text, possibly for a summary or abstract.
- A table with 3 columns and 10 rows, likely for a list of contents or a table of data.
- A footer section with dotted lines for text entry.

Other pages

Other pages layout template showing multiple sections of text and form fields.

Form fields include:

- Multiple sections of text with dotted lines for entry.
- Horizontal lines for separating sections.
- A table with 3 columns and 10 rows, similar to the one on the front page.
- Additional text blocks and form fields for detailed content.

What to expect

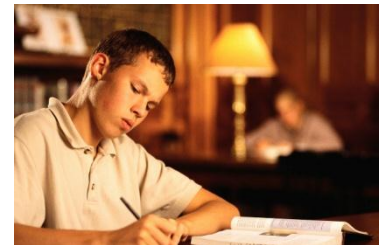
Instructions to Candidates

1. This is a CLOSED BOOK examination.
2. Simple calculators are allowed. Electronic devices, apart from simple calculators, are not permitted.
3. The paper comprises SEVEN questions each with multiple parts.
4. ANSWER FIVE QUESTIONS in the booklet, in the spaces provided.
5. Questions are worth equal marks. The mark to be awarded for each part is indicated. Marks total 5 * 10 marks each.
6. Additional space is available for rough work or extended answers on the back page of the booklet if required. Label any additional work carefully.
7. Take care to write LEGIBLY – it can be marked if it cant be read.
8. Write your final answers in INK, not pencil.
9. Make sure you write the answers you attempt in the “Answers area”
10. No bullet points for complete answer – there has to be an analysis
11. Mark the question in he front sheet you have answered.



Examination strategy

- Be aware of TIMING
 - 50 marks to be earned in 120 minutes
 - If question is worth 10 marks
 - Use space provided as a guide
- Read each question carefully
- Decide which topic the question is related to
- Show that you have assimilated the material delivered in THIS course
- Think, not just memorise
- Explain the answer not just bullet points



Thank you for participating in this
course

Make sure you complete the end
of course survey

Constructive comments will be
looked into

Good luck in your Careers !



I might see you in the future!
I might hire you !