ISYS2160
Information Systems in the Internet Age

Lecture 1: Introduction and Systems Approach/Socio-Technical Systems

Prof. Joseph Davis

School of Information Technologies

THE UNIVERSITY OF SYDNEY

Teaching Staff

- Dr. Na LIU (Lecturer and UoS Coordinator)
  - Email: liu.na@sydney.edu.au
  - Phone: 8627 4277
  - Office: SIT 442
- Prof. Joseph Davis (Lecturer)
  - Email: Joseph.davis@sydney.edu.au
- Ms. Niku Gorji (Teaching Assistant)
  - Email: nikugorji@gmail.com
- Mr. Adi Wibowo (Tutor)
  - Email: awib4756@uni.sydney.edu.au
- Ms. Himsha Sood (Tutor)
  - Email: hsoo4712@uni.sydney.edu.au

Objectives

- To understand the fundamental concepts related to information systems
- the nature, characteristics, and different classes of information systems (IS)
- To appreciate the strategic role of IS/IT in organizations, industries and society
- This UoS will provide a comprehensive conceptual and practical introduction to information systems (IS) in contemporary organisations.
  - How the key components of IS (i.e. people, software, hardware, data, and communication technologies) can be integrated and managed to create competitive advantage.
  - How information is used in organisations and how IT enables improvement in quality, speed, and agility.
  - Introduction to systems and development concepts, technology acquisition, and various types of application software that have become prevalent or are emerging in modern organisations and society.

About ISYS2140

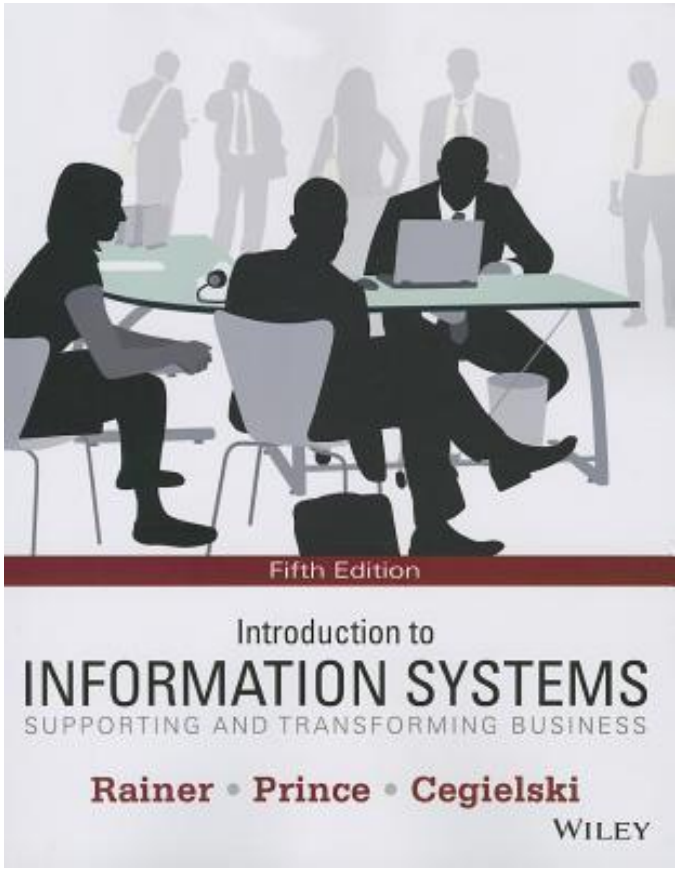
- Key topics covered:
  - Basic concepts of information systems
  - Systems thinking, systems approach, and socio-technical systems
  - Internet and E-business
  - E-payment and M-commerce
  - Online marketing and social media
  - Data, data management, and business intelligence
  - Information systems within in organization
  - Supply chain management and customer relationship management systems
  - Information systems for competitive advantage
  - Information systems development and acquisition
  - Information systems security
  - Information systems ethics

Assessment

- In-semester Assessment 50 marks
  - Tutorial Exercises (10 marks)
  - Mid-semester Quiz (Week 7) (20 marks)
  - Group Project (Week 13) (20 marks)
- Final Exam 50 marks
- Passing Criteria
  - In-semester Assessment ≥ 20 marks (40% of 50 marks)
  - Final Exam ≥ 20 marks (40% of 50 marks)
  - Overall ≥ 50 marks (50% of 100 marks)

Recommended Textbook

- Rainer, R.K., Prince, B., and Cegielski, C.G. *Introduction to Information Systems: Supporting and Transforming Business* (5th Ed). Wiley, 2014.
- Available in the university library
- Additional Readings (see course outline)



Course Policies

- Lectures
  - Start punctually at 12.05 pm. Do try and be in the lecture theatre by 11 am.
  - A short10-minute break after 50 minutes at 12.55 pm
  - Please silence your mobile phones

- Tutorials
  - Please ensure that you are in the class before the start of the lectures and tutorials
  - Please attend only your registered tutorial group
  - Project team will be formed within the tutorial groups
  - Project team formation will be finalized on Week 3 Tutorial

Course Policies

- Communication
  - Please check the Learning Management System (LMS) regularly
  - Email lecturer/tutors using your university email account, indicating ISYS2160 in the subject line
- Late assessments
  - Suppose you hand in work after the deadline:
  - If you have not been granted special consideration or arrangements
    - A penalty of 20% of the marks scored will be applied each day (or part) late
  - *Eg. your work would have scored 60% and is submitted 1 hour late*
    - *you will get 40%*
  - *Eg your work would have scored 70% and is 28 hours late*
    - *you will get 30%*
- Plagiarism
  - The University takes a serious view of plagiarism or any form of cheating
  - Please ensure **you cite** all sources of your work in written assignments.

Penalties for plagiarism

- The penalties are severe and include:

  1) a permanent record of academic dishonesty, plagiarism and misconduct in the University database and on your student file

  2) mark deduction, ranging from 0 for the assignment to Fail for the course

  3) expulsion from the University and **cancellation of your student visa**

- Do not confuse legitimate co-operation and cheating! You can discuss the assignment with another student, this is a legitimate collaboration, but you cannot complete the assignment together – everyone must write their own code or report, unless the assignment is group work.
- When there is copying between students, note that both students are penalised – the student who copies and the student who makes his/her work available for copying

Academic dishonesty and plagiarism

- Please read the University policy on Academic Honesty carefully:

  http://sydney.edu.au/elearning/student/EI/academic_honesty.shtml
- All cases of academic dishonesty and plagiarism will be investigated
- There is a new process and a centralized University system and database
- Three types of offenses:
  - Plagiarism – when you copy from another student, website or other source. This includes copying the whole assignment or only a part of it.
  - Academic dishonesty – when you make your work available to another student to copy (the whole assignment or a part of it). There are other examples of academic dishonesty.
  - Misconduct - when you engage another person to complete your assignment (or a part of it), for payment or not. This is a very serious matter and the Policy requires that your case is forwarded to the University Registrar for investigation.

Detection

- We will use the similarity detection software TurnItIn and MOSS to compare your assignments with these of other students (current and previous) and the Internet
  - Turnitin is for text documents: http://www.turnitin.com/en_us/higher-education
  - MOSS is for programming code: https://theory.stanford.edu/~aiken/moss/
- These tools are extremely good!
  - e.g. MOSS cannot be fooled by changing the names of the variables or changing the order of the conditions in `if-else` statements
- Examples of plagiarism in programming code:
  - http://www.upenn.edu/academicintegrity/ai_computercode.html

Student excuses

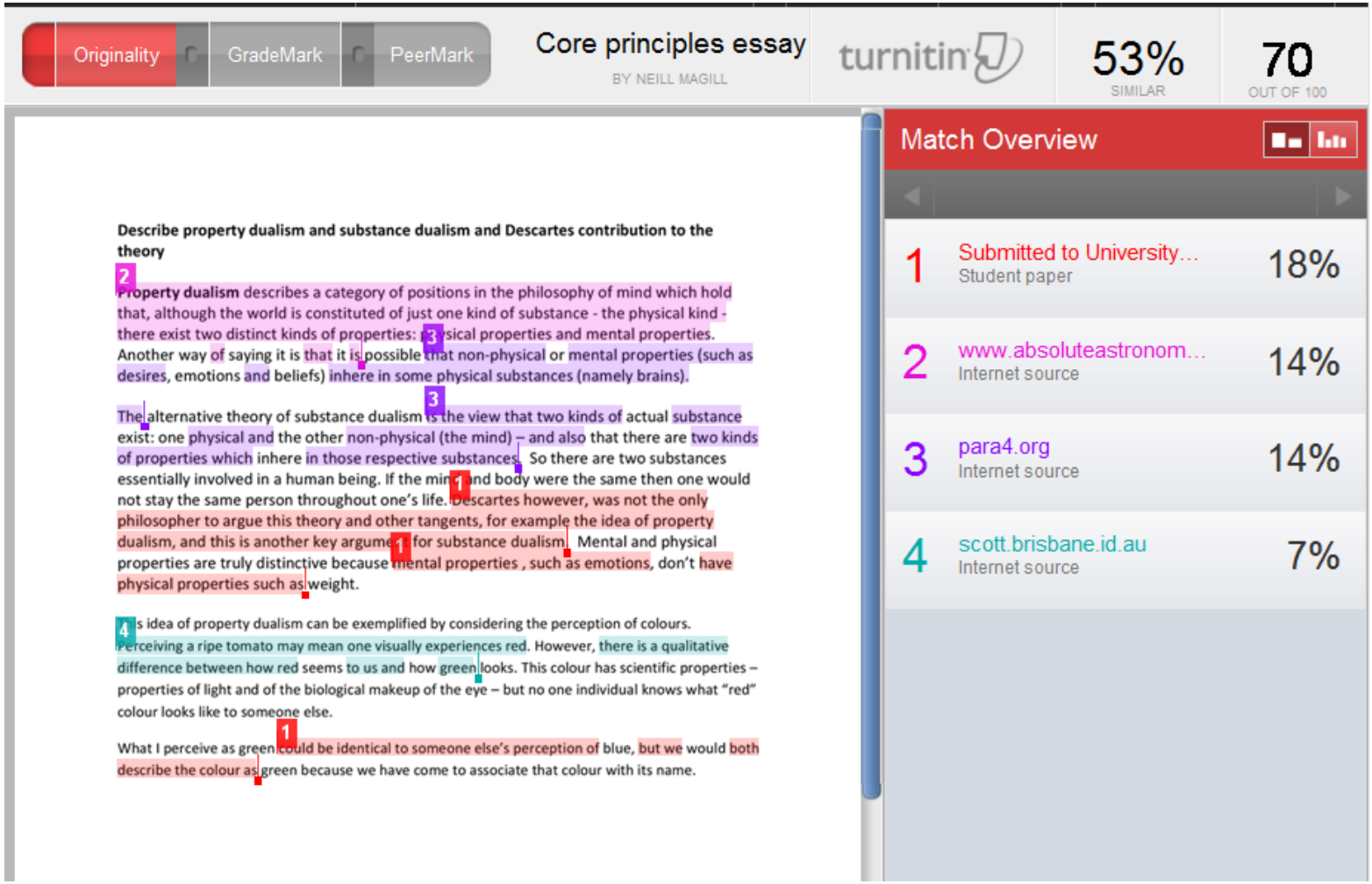
- All these are cases of plagiarism and academic dishonesty we have seen in our school and the student excuses are not acceptable:
  - *I sat the test and then posted the questions and solutions to my friends whose test was later in the week. I only wanted to help them understand the concepts that are examinable.*
  - *I posted parts of my code on my web page (group discussion forum) because my solution was cool (or I wanted to help them). I didn't expect them to copy it.*
  - *I tried to do the assignment on my own but I had problems with the extension part that I couldn't fix, so I submitted my core part and his extension part. I didn't cheat.*
  - *I finished my assignment but my friend had family problems. I felt sorry for her, so I gave her my assignment as an example. She said she only wanted to have a look and promised not to copy it.*
  - *The test has finished but the tutor hasn't collected the papers yet. I showed my answer to my friend. I didn't expect him to copy it.*
  - *He is my best friend. I had no choice but to let him copy my assignment.*

Key message

- Plagiarism and any form of academic dishonesty will be dealt with, and the penalties are severe
- We use plagiarism detection systems such as MOSS and TurnItIn that are extremely good. If you cheat, the chances you will be caught are very high.
- If someone asks you to see or copy your assignment, or to complete the assignment instead of them, just say: *I can't do this. This is against the University policy. I will not risk my future by doing this*.

Be smart and don't risk your future by engaging in plagiarism and academic dishonesty!

TurnitIn


Originality
GradeMark
PeerMark
Core principles essay
BY NEILL MAGILL
turnitin
53%
SIMILAR
70
OUT OF 100
Describe property dualism and substance dualism and Descartes contribution to the theory
Property dualism describes a category of positions in the philosophy of mind which hold that, although the world is constituted of just one kind of substance - the physical kind - there exist two distinct kinds of properties: physical properties and mental properties. Another way of saying it is that it is possible that non-physical or mental properties (such as desires, emotions and beliefs) inhere in some physical substances (namely brains).
The alternative theory of substance dualism is the view that two kinds of actual substance exist: one physical and the other non-physical (the mind) – and also that there are two kinds of properties which inhere in those respective substances. So there are two substances essentially involved in a human being. If the mind and body were the same then one would not stay the same person throughout one's life. Descartes however, was not the only philosopher to argue this theory and other tangents, for example the idea of property dualism, and this is another key argument for substance dualism. Mental and physical properties are truly distinctive because mental properties , such as emotions, don't have physical properties such as weight.
This idea of property dualism can be exemplified by considering the perception of colours. Perceiving a ripe tomato may mean one visually experiences red. However, there is a qualitative difference between how red seems to us and how green looks. This colour has scientific properties – properties of light and of the biological makeup of the eye – but no one individual knows what "red" colour looks like to someone else.
What I perceive as green could be identical to someone else's perception of blue, but we would both describe the colour as green because we have come to associate that colour with its name.
Match Overview
1 Submitted to University... Student paper 18%
2 www.absoluteastronom... Internet source 14%
3 para4.org Internet source 14%
4 scott.brisbane.id.au Internet source 7%

Course Policies

- Assessment: Coursework Policy 2014 (Part 14: Assessment)
  - http://sydney.edu.au/policies/showdoc.aspx?recnum=PDOC2014/378&RendNum=0
  - Cover Sheet for Group Project
    - http://sydney.edu.au/engineering/it/current_students/undergrad/policies/assignment_sheet_group.pdf
- Academic Dishonesty and Plagiarism :
  - http://sydney.edu.au/policies/showdoc.aspx?recnum=PDOC2012/254&RendNum=0
  - Tutorial on Plagiarism and Academic Honesty
    - http://sydney.edu.au/library/skills/elearning/learn/plagiarism/
  - Tutorial on Referencing
    - http://sydney.edu.au/library/skills/elearning/learn/referencing/
- Special Considerations:
  - notify coordinator by email *as soon as anything begins to go wrong*
  - http://sydney.edu.au/current_students/special_consideration/
- For any other policy documents, go to http://sydney.edu.au/policies/

School of Information Technologies

THE UNIVERSITY OF SYDNEY

General Housekeeping – Use of Labs

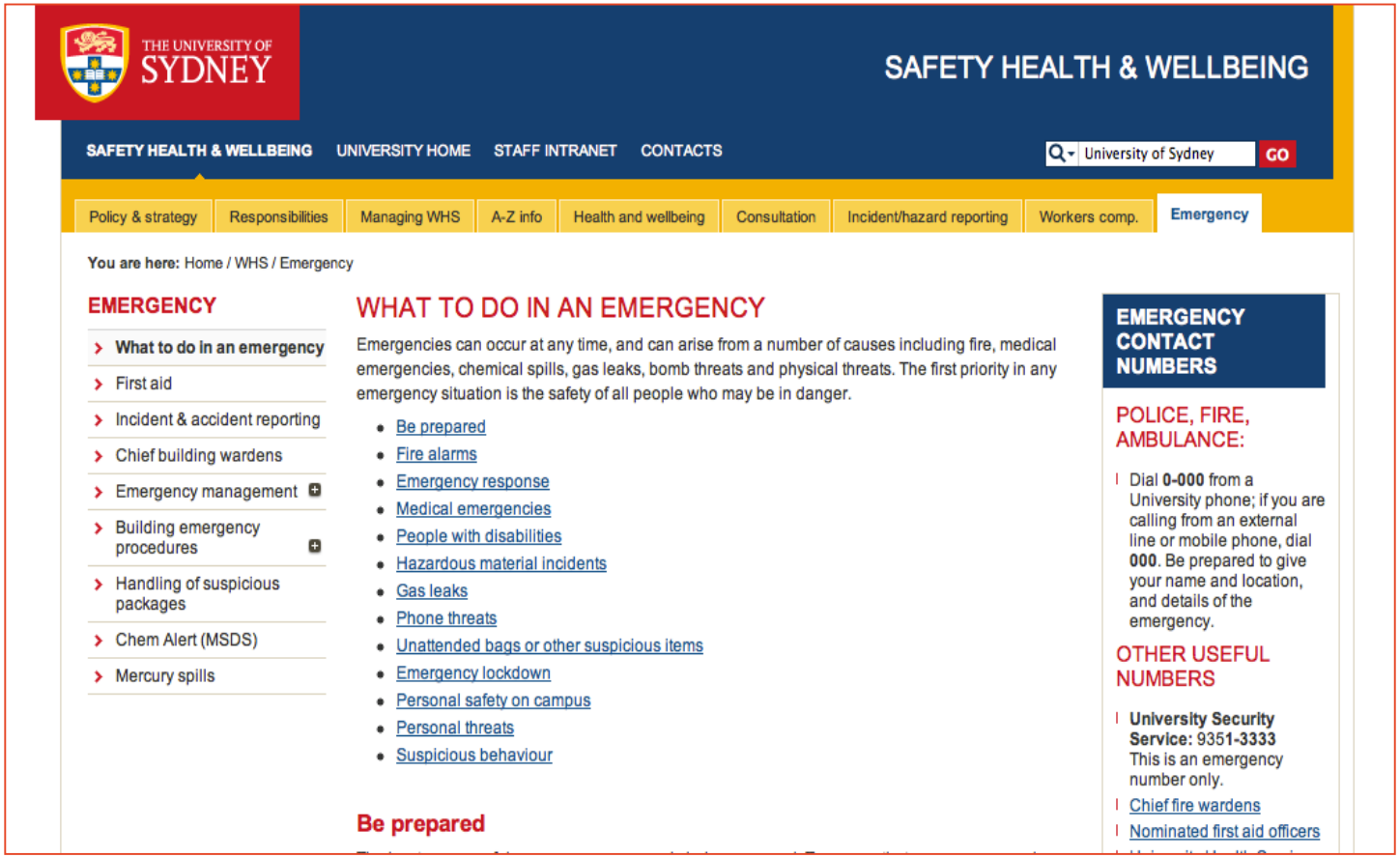
- Keep work area clean and orderly
- Remove trip hazards around desk area
- No food and drink near machines
- No smoking permitted within University buildings
- Do not unplug or move equipment without permission





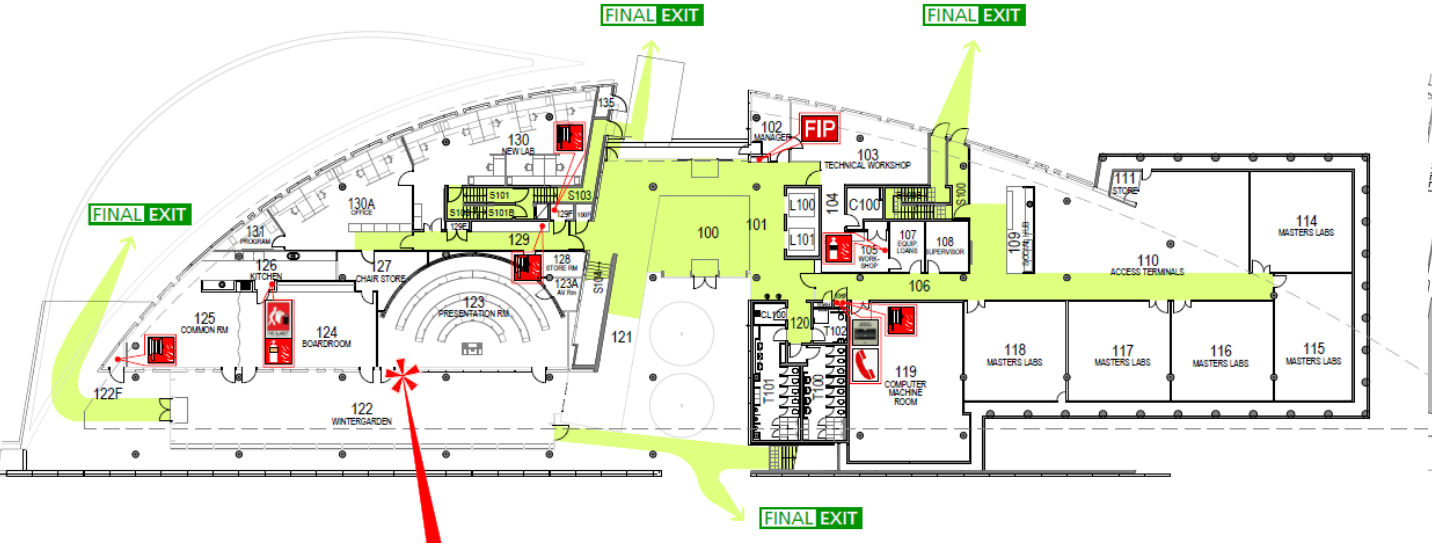
EMERGENCIES – Be prepared

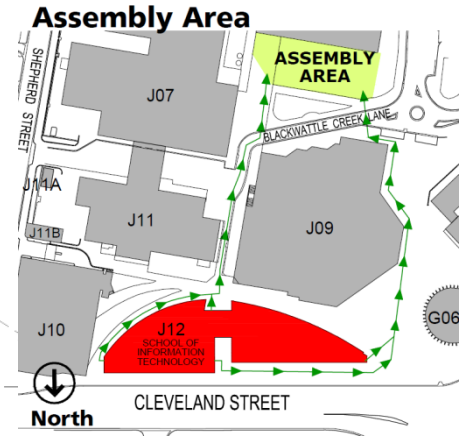
www.sydney.edu.au/whs/emergency



EMERGENCIES

WHERE IS YOUR CLOSEST SAFE EXIT ?





EMERGENCIES

Evacuation Procedures

ALARMS

BEEP... BEEP... Prepare to evacuate

1. Check for any signs of immediate danger.
2. Shut Down equipment / processes.
3. Collect any nearby personal items.

WHOOP... WHOOP... Evacuate the building

1. Follow the EXIT exit signs.
2. Escort visitors & those who require assistance.
3. DO NOT use lifts.
4. Proceed to the assembly area.

EMERGENCY RESPONSE

1. Warn anyone in immediate danger.
2. Fight the fire or contain the emergency, if safe & trained to do so.

If necessary...

3. Close the door, if safe to do so.
4. Activate the "Break Glass" Alarm  or 
5. Evacuate via your closest safe exit. EXIT 
6. Report the emergency to 0-000 & 9351-3333

MEDICAL EMERGENCY

- If a person is seriously ill/injured:
  1. **call an ambulance 0-000**
  2. **notify the closest Nominated First Aid Officer**

     If unconscious– send for Automated External Defibrillator (AED) AED **locations**.

     NEAREST to SIT Building (J12)
     - Electrical Engineering Building, L2 (ground) near lifts
     - Seymour Centre, left of box office
     - Carried by all Security Patrol vehicles

  3. **call Security -** 9351-3333
  4. **Facilitate the arrival of Ambulance Staff (via Security)**

Nearest Medical Facility
University Health Service in Level 3, Wentworth Building

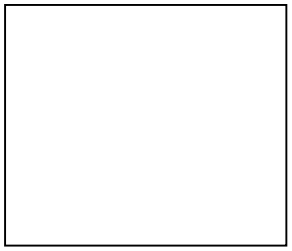
First Aid kit – SIT Building (J12)
kitchen area adjacent to Lab 110

School of IT Safety Contacts

CHIEF WARDEN
Name: Greg Ryan
Mobile:

FIRST AID OFFICERS

Name: Will Calleja
Location: 1 West
Phone: 9036 9706



Name: Katie Yang
Location: 2E-227
Phone: 9351 4918

Orally REPORT all INCIDENTS & HAZARDS to your SUPERVISOR

OR

Undergraduates: to Katie Yang 9351 4918

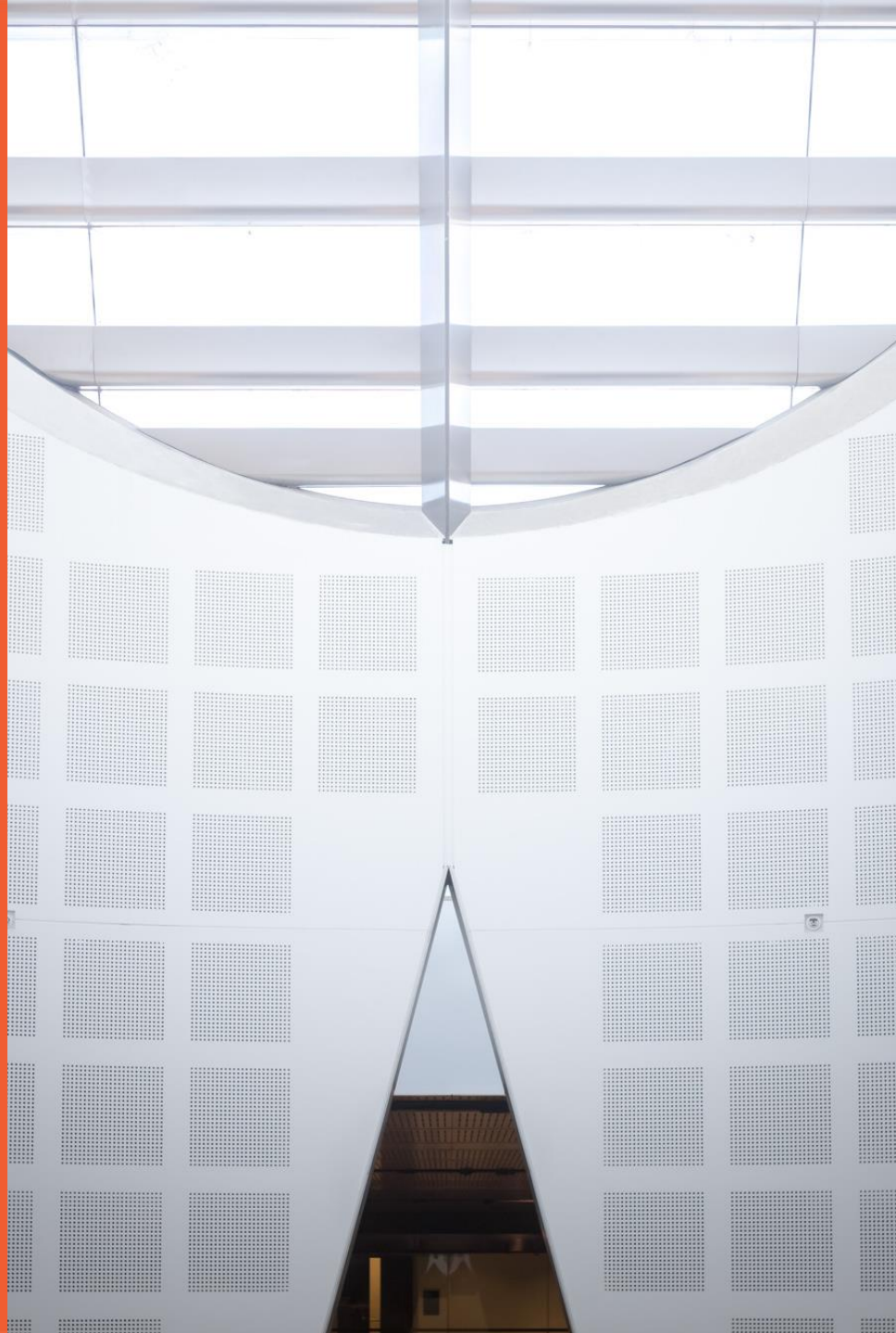
Coursework Postgraduates: to Cecille Faraizi 9351 6060

SIT School Manager: Shari Lee 9351 4158

Systems Approach, Systems Thinking, and Socio-technical systems

Prof. Joseph Davis

THE UNIVERSITY OF SYDNEY



What is a System?

A 'system' is set of inter-related components (parts) or subsystems.

We generally deal with 'teleological' (purposeful) systems.

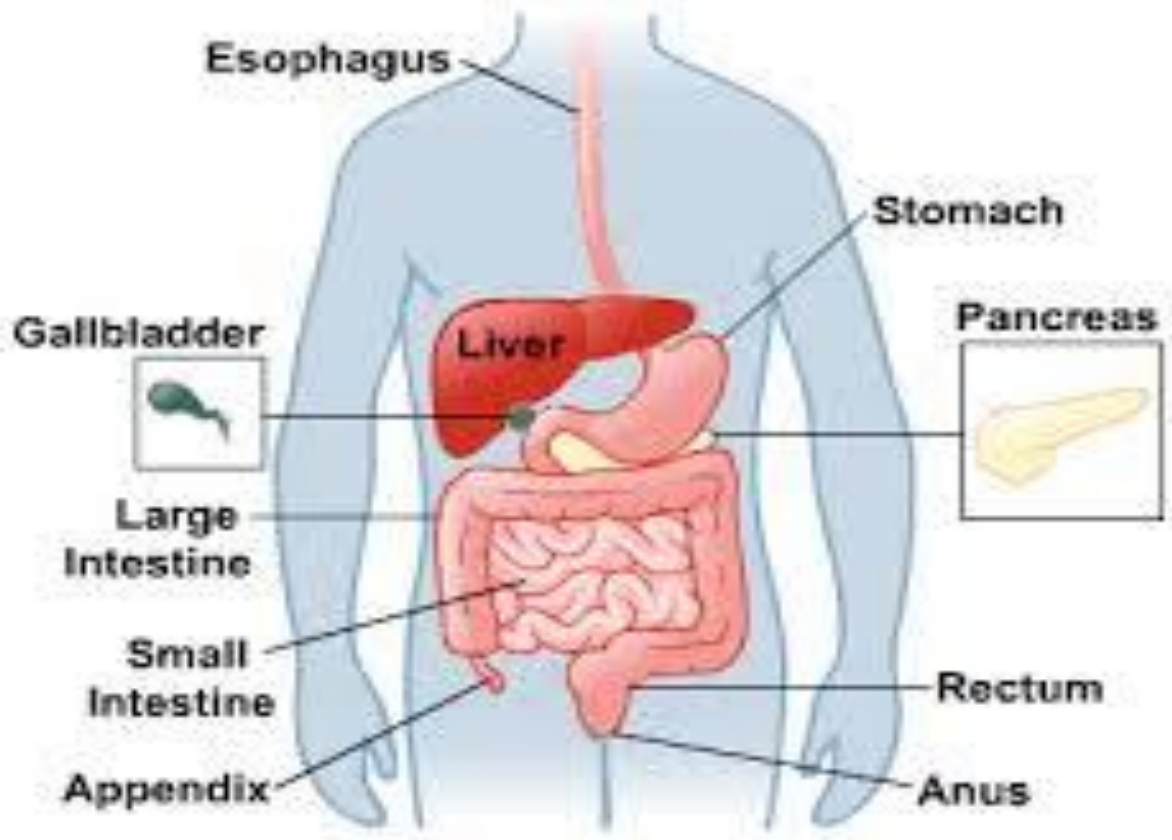
For teleological systems, we modify the definition:

A 'system' is made up of inter-related components that work together to achieve the overall objectives of the whole system


Applications
APP
APP
APP
APP
Monitor
Printer
Hard Drive
Operating System
Mouse
Keyboard

Software System

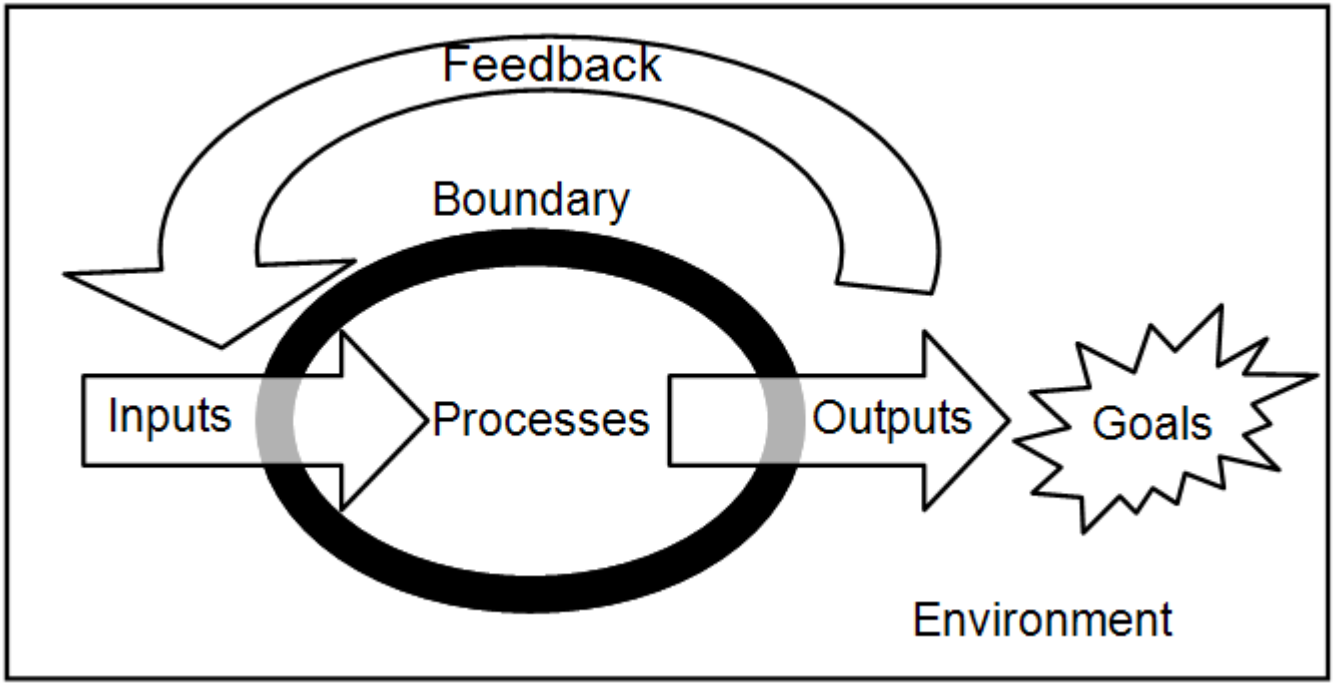
- A **software system** is a system of intercommunicating components based on software forming part of a computer system (a combination of hardware and software). It "*consists of a number of separate programs, configuration files, which are used to set up these programs, system documentation, which describes the structure of the system, and user documentation, which explains how to use the system*".
- While a computer program is a set of instructions (source, or object code) a software system has many more components such as specification, test results, end-user documentation, maintenance records, etc

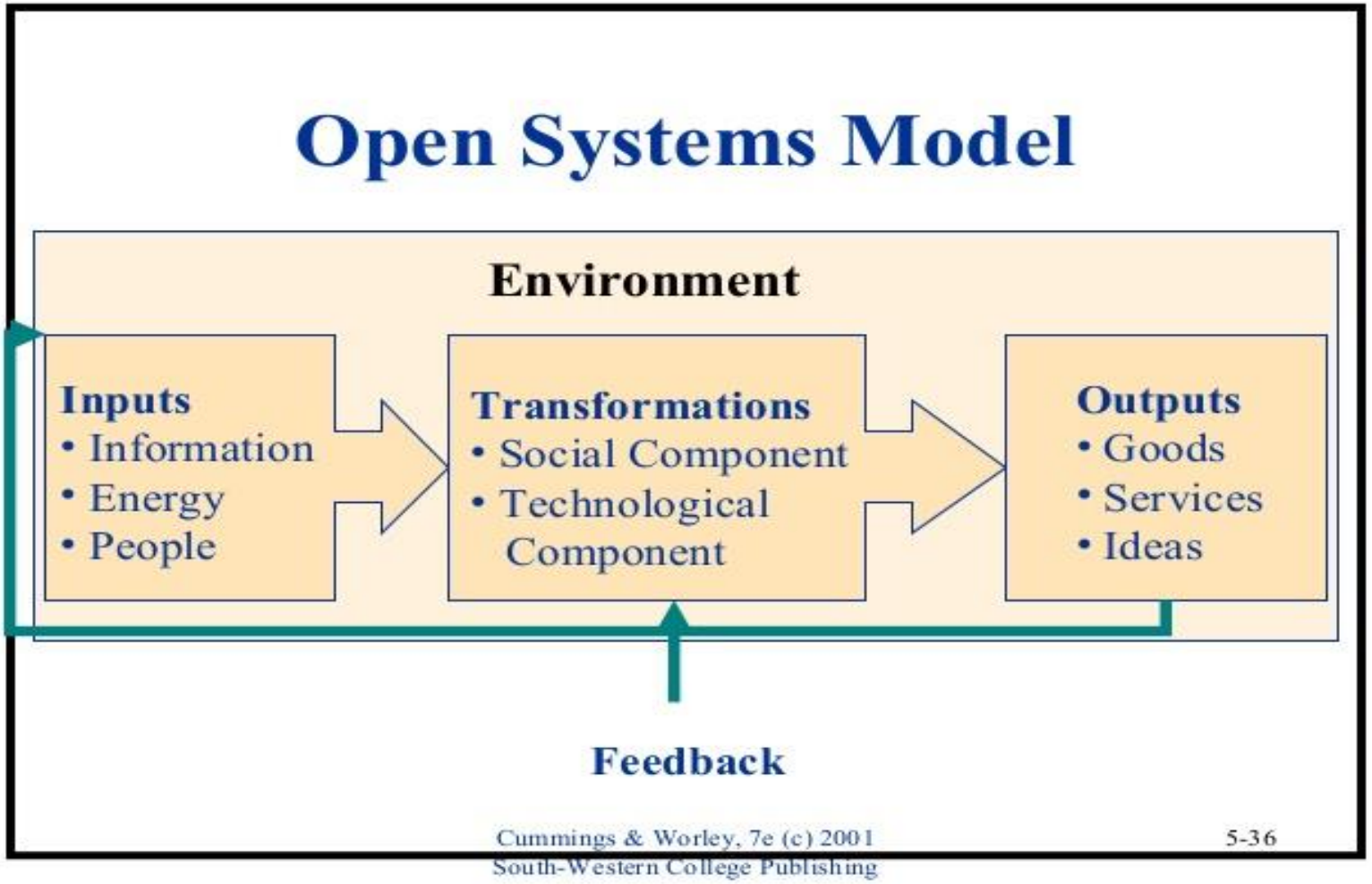

Esophagus
Stomach
Gallbladder
Liver
Pancreas
Large Intestine
Small Intestine
Rectum
Appendix
Anus

Process View of Systems

Input-Output Approach

- Inputs to the system (from the ext. envt.)
- Throughput processing/Transformation
- Outputs of the system (to the ext. envt.)
- Feedback Mechanisms
- Modelling of the overall system with a view to developing optimal solution.


Feedback
Boundary
Inputs
Processes
Outputs
Goals
Environment


Open Systems Model
Environment
Inputs
• Information
• Energy
• People
Transformations
• Social Component
• Technological Component
Outputs
• Goods
• Services
• Ideas
Feedback
Cummings & Worley, 7e (c) 2001
South-Western College Publishing
5-36

Key System Principles

– 1. Openness:

- System behaviour can only be understood in relation to the external environment
- Distinction between the system and the evironment – systems boundary
- Controllable and uncontrollable variables
- Transactional environment and variables that can be influenced.
- Role of leadership and managing upward in purposeful systems

Key Systems Principles: Purposefulness

- Value-guided systems
- Role of understanding (why actors do what they do)
- Rational, emotional and cultural dimensions
- Reaction- response- action
- Adaptiveness
- Active Role of Choice

Key Systems Principles (contd.) Emergent Property

- Property of the whole that cannot be deduced from the properties of the parts
- Emergent properties as the product of complex interactions among several elements
- Interactions among five basic processes: throughput, decision making, learning and control, membership, and conflict management.
- Measurement system

Multidimensionality

- Multiple interacting dimensions
- Seemingly opposing tendencies not only co-exist to form a complementary relationship
- Plurality of structures and processes.

Counter-intuitiveness

- Actions intended to produce certain outcomes may generate opposite results.
- Beyond certain point, quantitative change can lead to qualitative change – difference in degree versus difference in kind
- Inflection Points

The Systems Approach

The systems approach or systems thinking is a method of analysing or thinking about complex systems from the perspective of the total system, the goals of the overall system, the individual components or subsystems, and the inter-relationships and inter-dependencies between the components.

System Theory

System Theory: the transdisciplinary study of the abstract organization of complex phenomena, independent of their substance, type, or spatial, or temporal scale of existence. It investigates both the principles common to all complex entities, and the (usually mathematical) models that can be used to represent them.

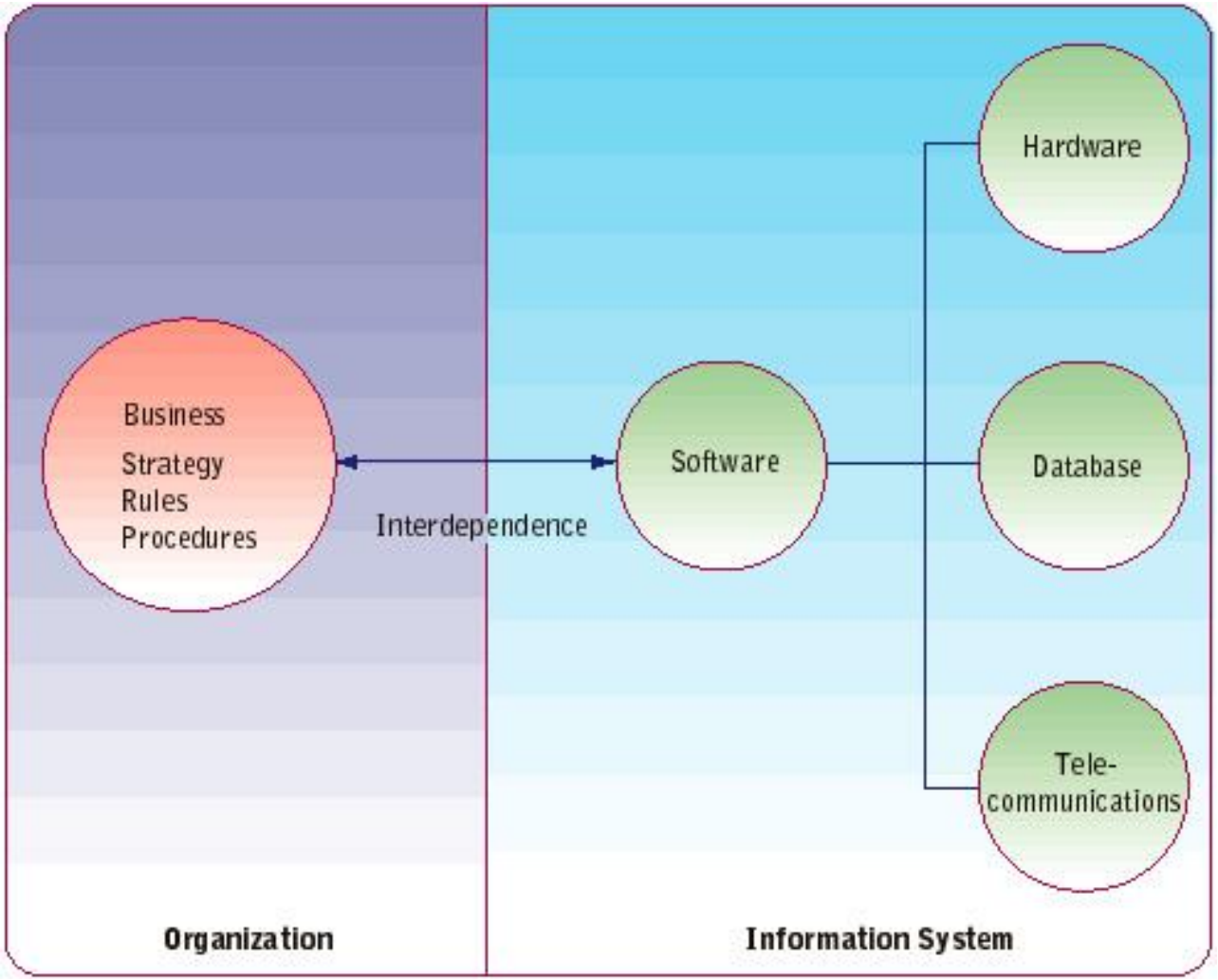
Types of Systems

- Teleological vs. ateleological systems
- Open vs. closed systems
- Mechanical Systems
- Biological/Living systems
- Social Systems
- Socio-technical systems

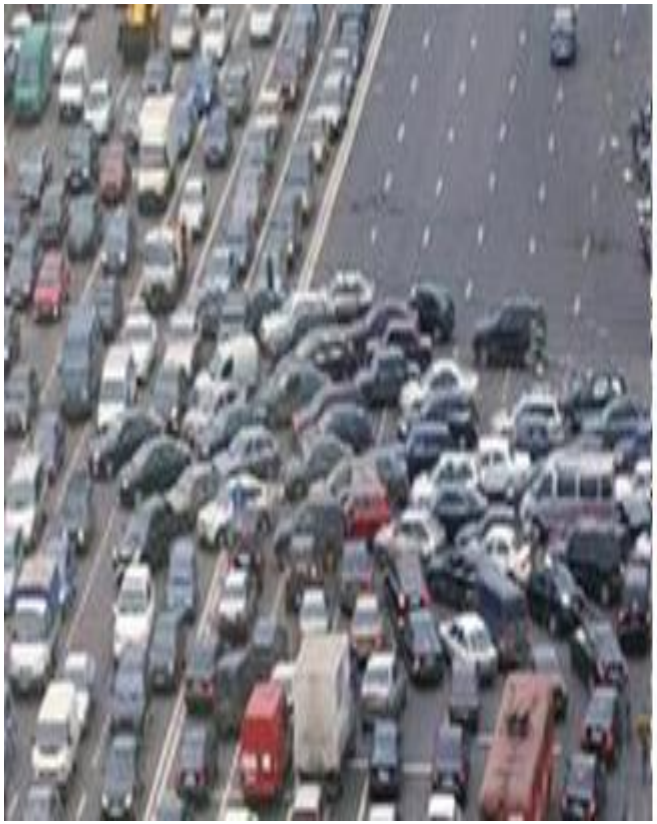
Socio-technical systems

Most Information Systems have a technical component and a social (human/organisational/societal) component which interact with each other in multiple ways.

Optimising the technical component without tweaking the social component is known to produce poor outcomes and results.



Socio-Technical Systems



Structure of Systems

- Overall objective represented as concrete measure(s) of performance,
- Inputs and Resources
- Process(es)
- Feedback Mechanisms
- Outputs
- System Boundary
- System Environment

Efficiency-oriented Approaches

- Concerned primarily with the efficiency of the system's operations,
- Has its origins in what is known as 'scientific management' or 'Taylorism'.
- Focus on eliminating all forms of 'waste' and 'slack' and eliminating them from the system to achieve cost savings.

General System Theory Approach

– Careful description of the total system including,

- full specifications of the parts or the components and their coordination,

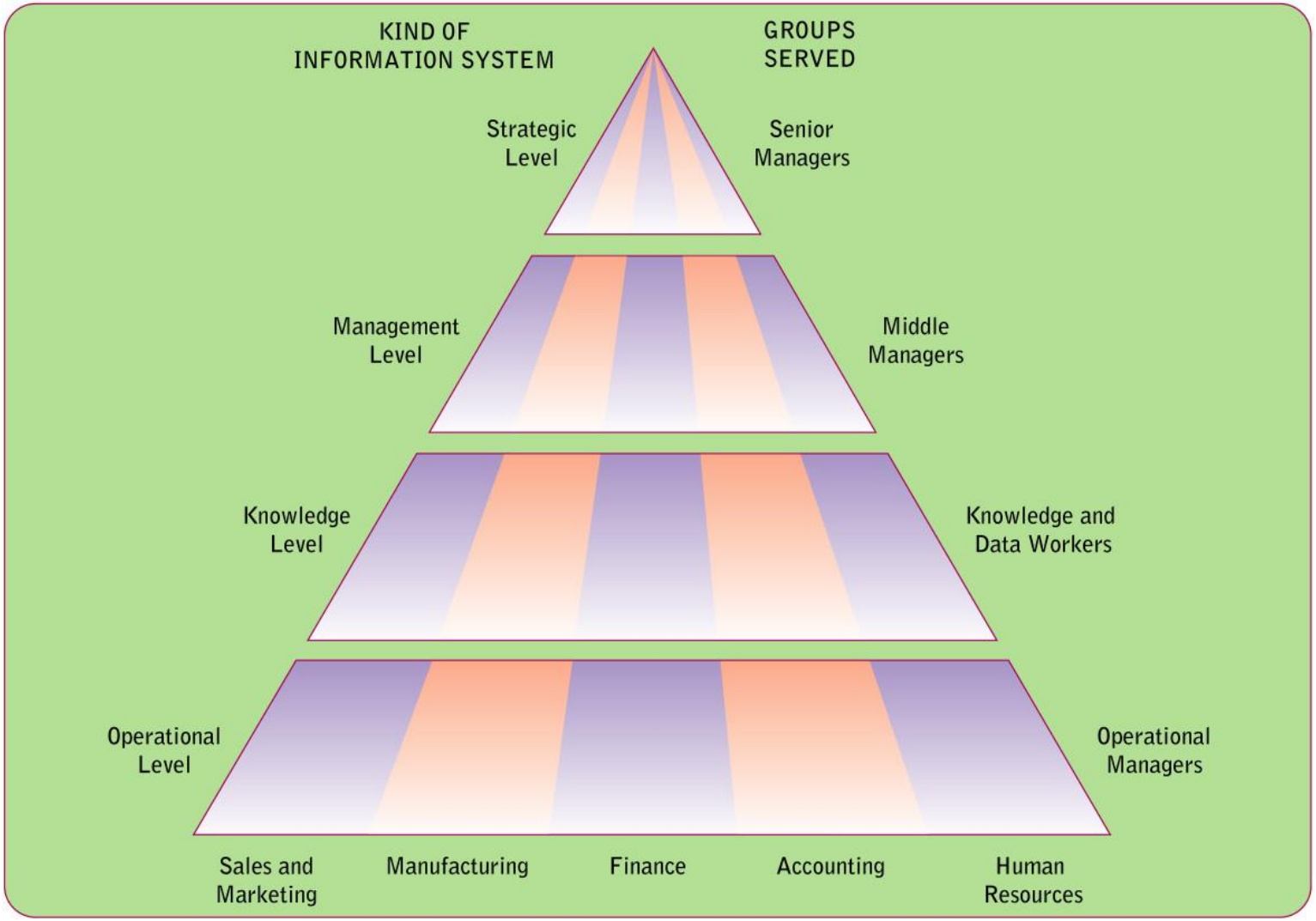
- identification of the measures of performance in measurable terms,

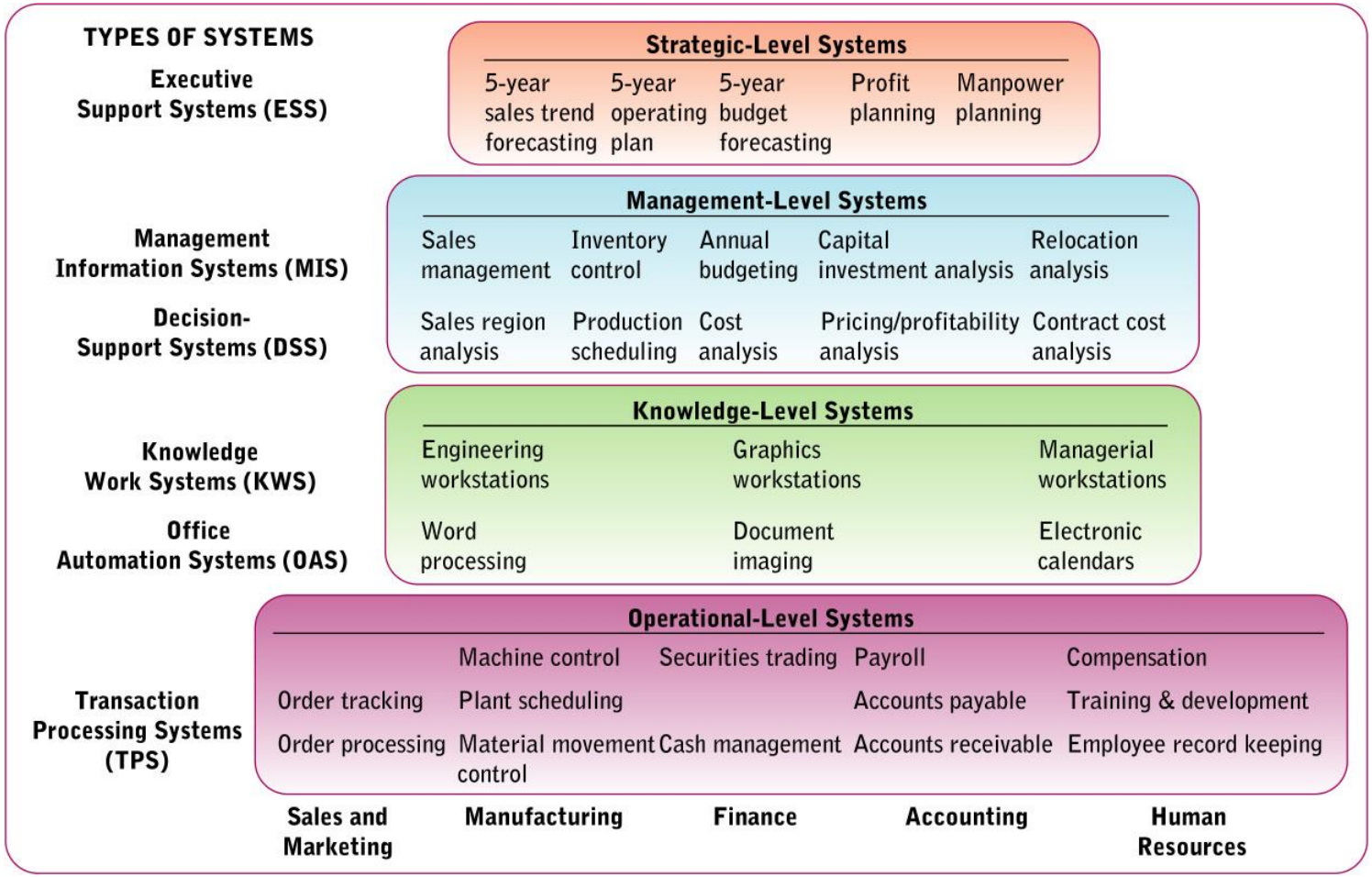
- definition of the system's boundaries and by implication, its external environment.

General System Theory-Based Approach

- Total system objectives – performance measures for the whole system,
- Resources available to the system – current and potential
- System's environment – the fixed constraints, what lies outside the system,
- The components of the system; their activities, goals, and measures of performance. How they mesh with the overall measures of performance,
- The management of the system

Organisational Levels




TYPES OF SYSTEMS
Executive Support Systems (ESS)
Strategic-Level Systems
5-year sales trend forecasting
5-year operating plan
5-year budget forecasting
Profit planning
Manpower planning
Management Information Systems (MIS)
Decision-Support Systems (DSS)
Management-Level Systems
Sales management
Inventory control
Annual budgeting
Capital investment analysis
Relocation analysis
Sales region analysis
Production scheduling
Cost analysis
Pricing/profitability analysis
Contract cost analysis
Knowledge Work Systems (KWS)
Office Automation Systems (OAS)
Knowledge-Level Systems
Engineering workstations
Graphics workstations
Managerial workstations
Word processing
Document imaging
Electronic calendars
Transaction Processing Systems (TPS)
Operational-Level Systems
Machine control
Securities trading
Payroll
Compensation
Order tracking
Plant scheduling
Accounts payable
Training & development
Order processing
Material movement control
Cash management
Accounts receivable
Employee record keeping
Sales and Marketing
Manufacturing
Finance
Accounting
Human Resources

TRANSACTION PROCESSING SYSTEMS (TPS)

- operational level
- inputs: transactions, events
- processing: updating
- outputs: detailed reports
- users: operations personnel

examples: accounts payable, payroll system, sales accounting system

TYPICAL TPS APPLICATIONS

Manufacturing & Production Systems

MAJOR FUNCTIONS OF SYSTEMS:

- Scheduling; Purchasing; Shipping / Receiving; Engineering; Operations

MAJOR APPLICATION SYSTEMS:

- Materials Resource Planning Systems; Purchase Order Control Systems; Engineering Systems; Quality Control Systems

TYPICAL TPS APPLICATIONS

Finance & Accounting Systems

MAJOR FUNCTIONS OF SYSTEMS:

- Budgeting, Billing, Cost Accounting

MAJOR APPLICATION SYSTEMS:

- General Ledger; Accounts Receivable / Payable; Budgeting; Funds Management Systems

*

TYPICAL TPS APPLICATIONS

Human Resources Systems

MAJOR FUNCTIONS OF SYSTEMS:

- Personnel Records; Benefits; Compensation; Labor Relations; Training

MAJOR APPLICATION SYSTEMS:

- Payroll; Employee Records; Benefit Systems; Personnel Training Systems

*

TYPICAL TPS APPLICATIONS

Other Types (e.g., University)

MAJOR FUNCTIONS OF SYSTEMS:

- Admissions; Grade Records; Student enrolments and Course Records

MAJOR APPLICATION SYSTEMS:

- Student Enrollment System; Student Transcript System; Class and tutorial Control System; Alumni Benefactor System

*

KNOWLEDGE WORK SYSTEMS (KWS)

- knowledge level
- inputs: design specs/documents
- processing: modelling
- outputs: designs, graphics, documents, reports
- users: technical and professional staff

MANAGEMENT INFORMATION SYSTEMS (MIS)

- management level
- inputs: high volume data
- processing: using simple models
- outputs: summary reports
- users: middle managers

example: annual budgeting

MANAGEMENT INFORMATION SYSTEMS

- structured & semi-structured decisions
- report control oriented
- past & present data
- internal orientation

*

DECISION SUPPORT SYSTEMS (DSS)

- management level- centred on specific decisions or class of decisions
- inputs: data from data warehouse/ multiple databases
- processing: interactive; use of GUIs and other easy-to-use user interfaces
- use of analytical and other models
- outputs: decision analysis
- users: professionals, staff

DECISION SUPPORT SYSTEMS (DSS)

- flexible, adaptable, quick
- user controls inputs/outputs
- Focus on support for decision processes (not automating!)
- sensitivity/what-if analysis capabilities
- sophisticated modeling tools and user interfaces

*