Unit 1-4.pdf

Unit 5-11.pdf

Unit 12-19.pdf

Unit-1

1. Introduction to Management Support Systems

1.1 Managers and decision making process

- Decision: A decision is a choice between two or more alternatives. If you only have one alternative, you do not have a decision. A typical thesaurus might use words like accommodation, agreement, arrangement, choice, compromise, declaration, determination, outcome, preference, resolution, result, and verdict to try and give the concept of "decision" some dimension.

- Decision-making is one of the defining characteristics of leadership. It's core to the job description. Making decisions is what managers and leaders are paid to do. Yet, there isn't a day that goes by that you don't read something in the news or the business press that makes you wonder, "What were they thinking?" or "Who actually made that decision?"

- Manager and Decision: Being in a managerial role usually requires prompt decision making. Decision making process is the process by which managers respond to opportunities, threats, analyze all the available options and make a sound decision which is commensurate with the goals of the organization. The decision taken helps decide the further course of action. Decision making is one of the core responsibilities of a manager's job. There are other responsibilities as well which usually involve problem solving and work distribution. A manager must make informed decisions based on his expertise, technical knowledge and experience. Managers use a wide range of decision strategies, often changing these strategies from one situation to the next. The strategies lead to a wide variety of choices of varying quality, depending on the decision being made. Managers are equipped with a number of useful techniques for diagnosing problems, clarifying values and goals, structuring and modeling decisions, and gathering useful information. The three kinds of managerial roles include:
    - Interpersonal -- which include figure heads and leaders
    - Informational -- who receive and disseminate critical information
    - Decisional -- who initiate activities, handle disturbances, allocate resources and negotiate conflicts.

  Decisions made by top managers commit the total organization towards a particular course of action. Decisions made by lower level managers implement the strategic decisions of top managers in the operating areas of the organization. Top managers make Category II decisions. Operating managers make Category I decisions, while the middle managers supervises the making of Category I decisions and support the making of Category II decisions. The success of the decision taken is a function of the decision quality and decision implementation.

- Decision making Process: eight steps that add structure and simplicity to the Decision making process.
    - Recognize and identify the problem.: Decisions are responses to situations or problems that need addressing. Therefore is important to have a clear definition of what needs addressing before attempting to go further in the decision making process.
    - Consider the nature of the problem that you are trying to resolve: What is the type issue, problem, or situation you need to address? Is it problematic in terms of

creating an awkward situation between individuals, is it needed to change direction of a business? Why does the problem need a decision? What are the results you are hoping to achieve by this decision?

- Analyze or research the problem: It is important to gather all the information involved in the problem or question, so that informed choices can be made.
- Develop a list of possible solutions: List the possible decisions that could be made, and what their consequences would be.
- Select the best alternative: Look at the list drawn up in point 4 and choose the best solution for the situation.
- Execute the best choice: Sometimes the hardest part of making a decision is taking action. The best decisions are ones that deliver strong decision action.
- Follow Up and communication: A good decision needs to be followed throughout its process and constant communication made with those involved.
- Feedback: It is extremely important to gather feedback on a decision. This determines the overall success of and reaction to the decision.

1.2 The nature of manager’s works

The Canadian academic, Henry Mintzberg(1973), PhD thesis at the MIT Sloan School of Management *Analyzing the actual work habits and time management of chief executive officers (CEOs),* research involved observing and analyzing the activities of the CEOs of five private and semi-public organizations identified six characteristics of the job:

- Managers process large, open-ended workloads under tight time pressure - a manager's job is never done.
- Managerial activities are relatively short in duration, varied and fragmented and often self-initiated.
- CEOs prefer action and action driven activities and dislike mail and paperwork.
- They prefer verbal communication through meetings and phone conversations.
- They maintain relationships primarily with their subordinates and external parties and least with their superiors.
- Their involvement in the execution of the work is limited although they initiate many of the decisions.

Mintzberg then identified ten separate roles in managerial work, each role defined as an organized collection of behaviors belonging to an identifiable function or position. He separated these roles into three subcategories: interpersonal contact (1, 2, 3), information processing (4, 5, 6) and decision making (7-10).

S.N	function or position	Role
1	FIGUREHEAD	ceremonial and symbolic duties as head of the organisation
2	LEADER	adopts a proper work atmosphere and motivates and develops subordinates;
3	LIASION	develops and maintains a network of external contacts to gather information
4	MONITOR	gathers internal and external information relevant to the organisation
5	DISSEMINATOR	transmits factual and value based information to subordinates
6	SPOKESPERSON	communicates to the outside world on performance and policies
7	ENTREPRENEUR	designs and initiates change in the organisation
8	DISTURBANCE HANDLER	deals with unexpected events and operational breakdowns
9	RESOURCE ALLOCATOR	controls and authorizes the use of organizational resources
10	NEGOTIATOR	participates in negotiation activities with other organizations and individuals

He identified four clusters of independent variables: external, function related, individual and situational. He concluded that eight role combinations were 'natural' configurations of the job:

- contact manager -- figurehead and liaison
- political manager -- spokesperson and negotiator
- entrepreneur -- entrepreneur and negotiator
- insider -- resource allocator
- real-time manager -- disturbance handler
- team manager -- leader
- expert manager -- monitor and spokesperson
- new manager -- liaison and monitor

1.3 Need for computerized decision support

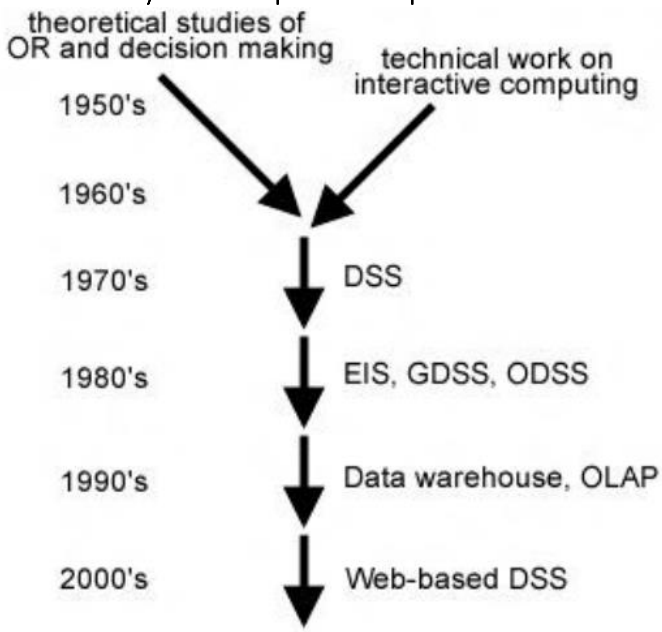
A Decision Support System (DSS) is a computerized system that assists in corporate decision making, with a decision being a choice between alternatives based on the estimated values of those alternatives. Generally, decision support systems are interactive, flexible, and adaptable information systems, developed to support the solution of non-structured management problems for improved decision making. These systems are designed to make use of data in order to help identify certain problems a business might be experiencing, and to help make the decisions necessary to address and work those problems out.Some of the benefits that Decision Support Systems bring to the Decision Making process are:

- Speed up the process of decision making
- Increases organizational control
- Encourages exploration and discovery on the part of the decision maker
- Speeds up problem solving in an organization
- Facilitates interpersonal communication
- Promotes learning or training

- Generates new evidence in support of a decision
- Creates a competitive advantage over competition
- Reveals new approaches to thinking about the problem space
- Helps automate managerial processes
- Decision Support Systems can be used by all sorts of businesses as assistive tools. They help in a variety of circumstances including problem management, data analysis and forecasting situations.

1.4 Decision support technologies:

- Types of Decision Support Systems:
    - Model-driven DSS puts emphasis on manipulation of a statistical, financial, or simulation model. This type of DSS uses data and parameters provided by users to assist decision makers in analyzing a situation; (they are not necessarily data intensive.) Parameters are provided by users for the analysis of a situation.
    - Communication-driven DSS supports more collaboration on a shared task. Examples include integrated tools like Microsoft's NetMeeting or Groove or SharePoint.
    - Data-driven DSS emphasizes manipulation of a chronological series of corporate internal data or occasionally, external data.
    - Document-driven DSS manages and manipulates unstructured information in from a variety of electronic formats.
    - Knowledge-driven DSS provides specialized problem solving expertise stored as facts, rules, procedures, or in similar structures.
    - Decision support systems often use Business Intelligence and Data Mining technology to provide aggregations of timely data as well as additional valuable insight. Data mining and predictive analytics often provide helpful information for decision support.



- ***Decision Support Technologies***
    - Management Support Systems (MSS): Subset of management information system (MIS), it extends the information retrieval capabilities of the end-users with 'query and analysis functions' for searching a database, generating 'what if' scenarios, and other such purposes.

- Decision Support Systems (DSS): Decision Support Systems often include features that allow the user to project how a business result would be affected if an underlying assumption were to change. For example, an investment decision might look less attractive if the company's cost of capital is driven up by an increase in the federal funds rate. The Decision Support System could calculate the cost and the return on investment in several scenarios.

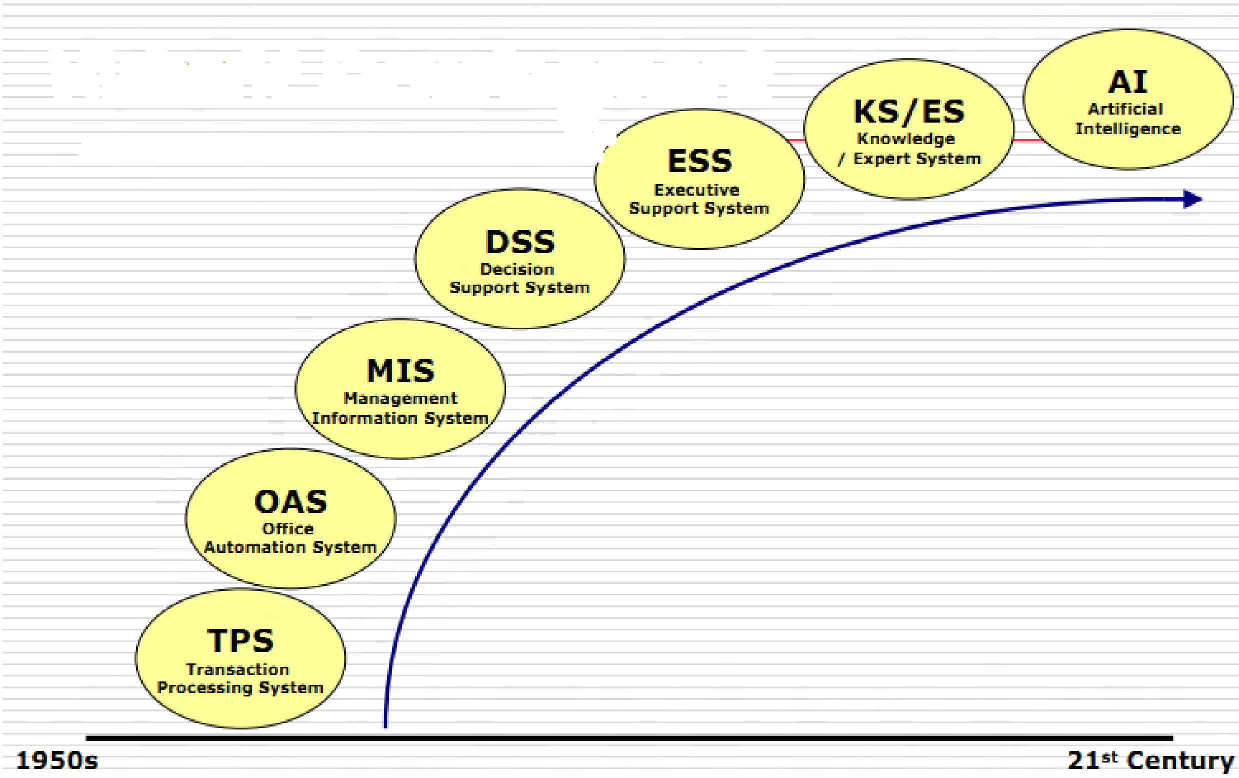
- Group Support Systems (GSS), including Group DSS (GDSS): Group Decision Support Systems (GDSS) are a class of electronic meeting systems, a collaboration technology designed to support meetings and group work .GDSS are distinct from computer supported cooperative work (CSCW) technologies as GDSS are more focused on task support, whereas CSCW tools provide general communication support .
- Executive Information Systems (EIS):Executive Information Systems (sometimes known as Scorecards or Dashboards, depending on their graphical presentation) are designed to deliver specific key information to top managers at a glance, with little or no interaction with the system. A decision support system is an analytical application that permits the user to call up information from the data warehouse and manipulate it to derive actionable information.
- Expert Systems (ES): An expert system is a computer system that emulates the decision-making ability of a human expert. Expert systems are designed to solve complex problems by reasoning about knowledge, like an expert, and not by following the procedure of a developer as is the case in conventional programming.

- Artificial Neural Networks (ANN): An artificial neural network (ANN), usually called neural network (NN), is a mathematical model or computational model that is inspired by the structure and/or functional aspects of biological neural networks. A neural network consists of an interconnected group of artificial neurons, and it processes information using a connectionist approach to computation.

- Hybrid Support Systems
- Cutting Edge Intelligent Systems
  (Genetic Algorithms, Fuzzy Logic, Intelligent Agents, ...)
-

1.5 Concept of MIS

MIS refers broadly to a computer-based system that provides managers with the tools for organizing, evaluating and efficiently running their departments with right Information to the right person at the right place at the right time in the right form at the right cost.
The three sub-componentsManagement, Information and System together bring out the focus clearly & effectively.



A management information system (MIS) provides information that is needed to manage organizations efficiently and effectively. Management information systems involve three primary resources: people, technology, and information or decision making. Management information systems are distinct from other information systems in that they are used to analyze operational activities in the organization. Academically, the term is commonly used to refer to the group of information management methods tied to the automation or support of human decision making, e.g. decision support systems, expert systems, and executive information systems.

Unit 2

2.1 System Concept:

2.1.1 Definition:

The word system means plan, method, order, and arrangement. A " system", says the dictionary, is a regularly interacting or inter – dependent group of items forming a united whole. A system is thus a set of interacting elements, interacting with each other to achieve a predetermined objective or goal. For example, in a computer system, the computer receives inputs and processes than produces the output.

A system is a group of interrelated components working together towards a common goal by accepting inputs and producing outputs in an organized transformation process.

IEEE defines software as, “the collection of computer programs, procedures, rules and associated documentation and data”. This definition clearly states that, the software is not just a collection of programs, but includes all associated documentation and data. This implies that, software development process should focus on all the things constitute the software.

2.1.2 Characteristics:

There are some common characteristics of any system.

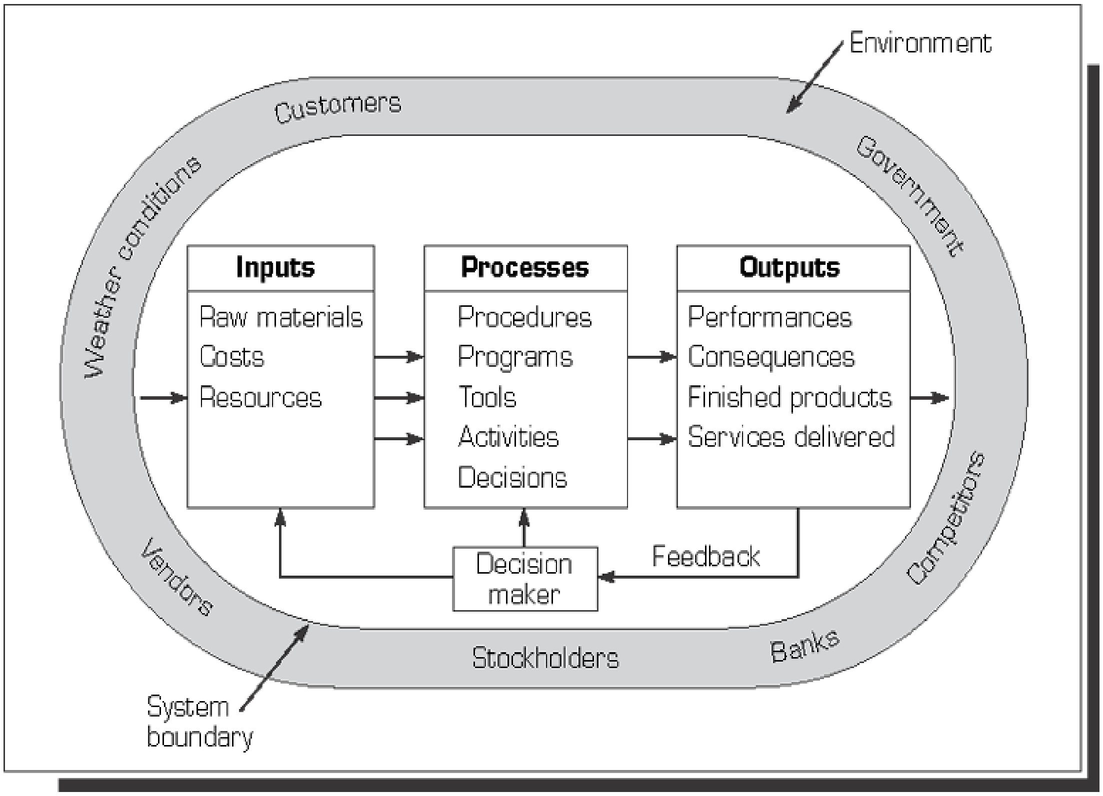
- Every system has a certain objectives and goals.
- Main system has a several subsystems or models.
- The structure of the system is representation of the interaction and interrelationships between different components or subsystems that from a system.
- The lifecycle of the system is expression of the phases in the alive usage life of the system.
- System operates in the terms of goals and predetermined scope.
- Systems in real life do not operate in isolation.

2.1.3 Elements of System:

It has five basic interacting components.

- **Input:**
  - Capturing / accepting and assembling components that enter the system to be processed *Example:* raw data, raw material etc.
- **Processing**:
  - Process that series of changes to be done on information, to convert input into output. *Example:* data processing, manufacturing process etc.
- **Output**:
  - Which produced by the transformation process to their ultimate destination. *Example:* reports, finished products etc.
- **Control :**
  - The control elements guide the system. It is the decision making sub system that controls the pattern of activities governing input, processing and output.
- **Feedback:** Control in a dynamic system is achieved by feedback. Feedback measures output against a standard in some form of cybernetic procedure that includes communication and control.

Figure 2.1 The System and Its Environment



2.1.4 **Types of System:**

- **PHYSICAL:** Systems are tangible entities that may be static or dynamic Example of static - Office desk, Chair Example of Dynamic - Programmed Computer.
- **ABSTRACT:** Systems are conceptual or non-physical entities. They may be as straight forward as formulas of relationships among set of variables or models is the abstract conceptualization of physical system.
- **OPEN & CLOSED SYSTEM**: A closed system is one which is self-contained. It has no interaction with its environment. No known system can continue to operate for a long period of time without interacting with its environment. An open system continuously, interacts with its environment. This type of system can adapt to changing internal and environmental conditions. A business organization is an excellent example of an open system.
- **Deterministic and Probabilistic System:** The behavior of a deterministic system is completely known. There is no uncertainty involved in defining the outputs of the system knowing the inputs. This implies that the interaction between various subsystems is known with certainty. Computer program is a good example of deterministic system, here, knowing the inputs, the outputs of the program can be completely defined. In the probabilistic system, the behavior cannot be predicted with certainty; only probabilistic estimates can

be given. In this case, the interactions between various subsystems cannot be defined with certainty.

- **MAN MADE INFORMATION SYSTEM: COMPUTER BASED INFORMATION SYSTEM:**

This category of the information system mainly depends on the computer for handling business applications. System analysts develop several different types of information systems to meet a variety of business needs. There is a class of systems known collectively as Computer Based Information System. Computer based information systems are of too many types. They are classified as:

- Transaction processing System (TPS):
    - A transaction processing system can be defined as a computer based system that capture, classifies, stores, maintains, updates and retrieves transaction data for record keeping. Transaction Processing system are aimed at improving the routine business activities on which all organisations depend.

- Management Information Systems (MIS):
    - MIS can be described as information system that can provide all levels of management with information essential to the running of smooth business. This information must be as relevant timely, accurate, complete, concise and economically feasible must be as relevant, timely, accurate, complete and concise as is economically feasible.

- Decision support system (DSS):
    - Decision support systems assist managers who must make decisions that are not highly structured, often called unstructured or semi-structured decisions. A decision is considered unstructured if there are not clear procedures for making the decision and if not all the factors to be considered in the decision can be readily identified in advance.

- Office Automation Systems (OAS):
    - Office automation systems are among the newest and most rapidly expanding computer based information systems. They are being developed with the hopes and expectations that they will increase the efficiency and productivity of office workers-typists, secretaries, administrative assistant, staff professionals, managers and the like. Many organisations have taken the first step toward automating their offices.
- Open System and Closed System
  A closed system is one which is self contained. It has no interaction with its environment. No known system can continue to operate for a long period of time without interacting with its environment. An open system continuously,

interacts with its environment. This type of system can adapt to changing internal and environmental conditions. A business organization is an excellent example of an open system.
Many mathematical models are confined to closed systems. A special type of closed system is called the black box. In such a system inputs and outputs are well defined but the process itself is not specified. DSS attempt to deal with systems that are fairly open. Such systems are complex, and during their analysis it is necessary to check the impacts on and from the environment.

Major differences between closed and open systems

Open System	Closed System
Open system interacts or communicates with the environment constantly	Whereas a closed system does not react with the environment
An open system has an infinite scope till the organization services	Whereas a closed system has limited shape
In an open system relevant variables keep on interacting.	Whereas the variables in a closed system are self contained
An open system is generally flexible and abstract	Whereas a closed system is rigid and mathematical

2.2 Models, Degree of model, Benefits of model

A major characteristic of DSS is the inclusion of a modeling capability. The basic idea is to execute the DSS analysis on a model of reality rather than on reality itself.

A model is a physical, mathematical, or otherwise logical representation of a system, entity, phenomenon, or process. Briefly, it is a simplified representation or abstraction of reality. It is usually simplified because reality is too complex to copy exactly and because much of the complexity is actually irrelevant to the specific problem. The characteristics of simplification and representation are difficult to achieve simultaneously in practice (they contradict each other). The representation of systems or problems through models can be done at various degrees of abstraction; therefore models are classified, according to their degree of abstraction, into three groups.

Iconic (Scale) Models. An iconic model , the least abstract model , is a physical replica of a system or graphical display that looks like the system being modeled, usually based on a different scale from the original. Iconic models may appear to scale in three dimensions, such as that of an airplane, car, bridge, or production line. Graphical user interface and object-oriented programming are other examples of the use of icons.

Analog Models. An analog model does not look like the real system but behaves like it. It is more abstract than an iconic model and is considered a symbolic representation of reality. There are usually two–dimensional charts or diagrams: that is they could be physical models, but the shape of the model differs from that of the actual system.

Mathematical (Quantitative) Models. The complexity of relationships in many organizational systems cannot be represented with icons or analogically, or such representation may be cumbersome and time-consuming. Therefore a more abstract model is used with the aid of mathematics. Most DSS analysis is executed numerically with the aid of mathematical or other quantitative models.

With recent advances in computer graphics, there is an increased tendency to use iconic and analog models to complement mathematical modeling in DSS.

- Iconic: Small physical replication of system
- Analog: Behavioral representation of system, May not look like system
- Quantitative (mathematical): Demonstrates relationships between systems

2.3 The Decision Making Process

Simon's original three phases:

- Intelligence
- Design
- Choice

He added fourth phase later:

- Implementation

Phase	Major Activities
Intelligence	Scan the environment ,Analyze organizational goals, Collect data, Identify problem, Categorize problem, Programmed and non-programmed, Decomposed into smaller parts, Assess ownership and responsibility for problem resolution
Design	Develop alternative courses of action, Analyze potential solutions, Create model, Test for feasibility , Validate results, Select a principle of choice, Establish objectives, Incorporate into models, Risk assessment and acceptance, Criteria and constraints
Choice	• Develop alternative courses of action • Analyze potential solutions • Create model • Test for feasibility • Validate results • Select a principle of choice – Establish objectives

	– Incorporate into models – Risk assessment and acceptance – Criteria and constraints • Principle of choice – Describes acceptability of a solution approach • Normative Models – Optimization • Effect of each alternative – Rationalization • More of good things, less of bad things • Courses of action are known quantity • Options ranked from best to worse – Sub-optimization • Decisions made in separate parts of organization without consideration of whole
Implementation	• Putting solution to work • Vague boundaries which include: – Dealing with resistance to change – User training – Upper management support

Figure 2.3 DSS Support

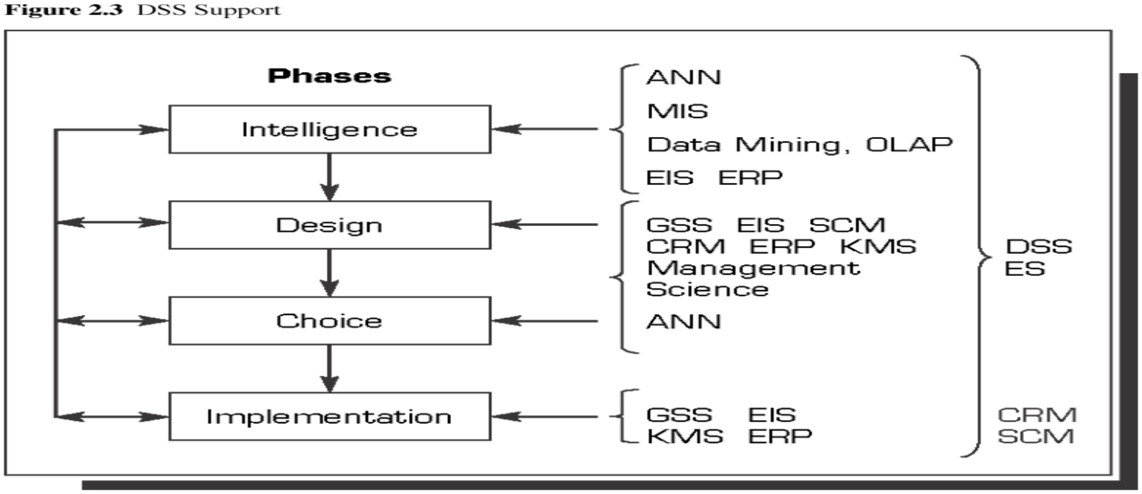
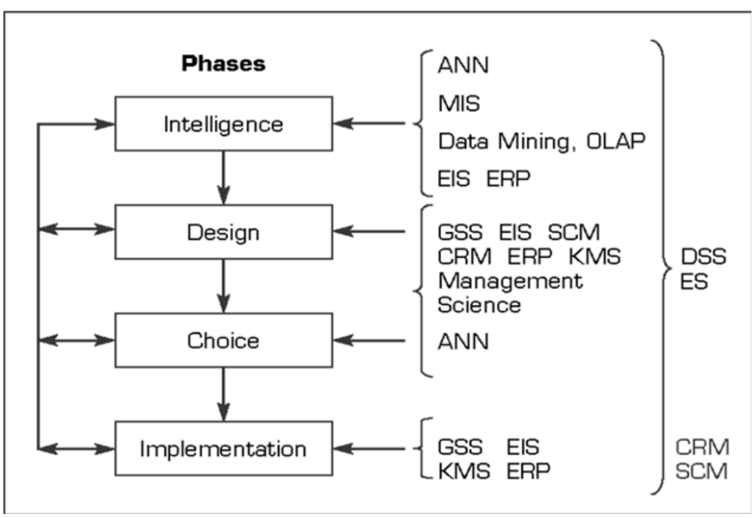


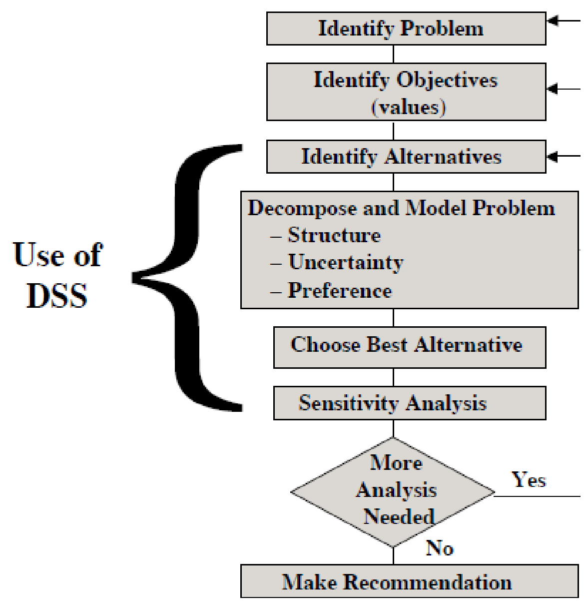
TABLE 2.4 Cognitive-style Decision Approaches

Problem-solving Dimension	*Heuristic*	*Analytic*
Approach to learning	Learns more by acting than by analyzing the situation and places more emphasis on feedback	Employs a planned sequential approach to problem solving; learns more by analyzing the situation than by acting and places less emphasis on feedback
Search	Uses trial and error and spontaneous action	Uses formal rational analysis
Approach to analysis	Uses common sense, intuition, and feelings	Develops explicit, often quantitative, models of the situation
Scope of analysis	Views the totality of the situation as an organic whole rather than as a structure constructed from specific parts	Reduces the problem situation to a set of underlying causal functions
Basis for inferences	Looks for highly visible situ-ational differences that vary with time	Locates similarities or common alities by comparing objects

MSS CATEGORIES BY PHASES OF DECISION-MAKING PROCESS

[Turban 2005] [Laskey 2006]





Unit 3

3 Overview of Decision Support System (DSS)

3.1 Introduction:

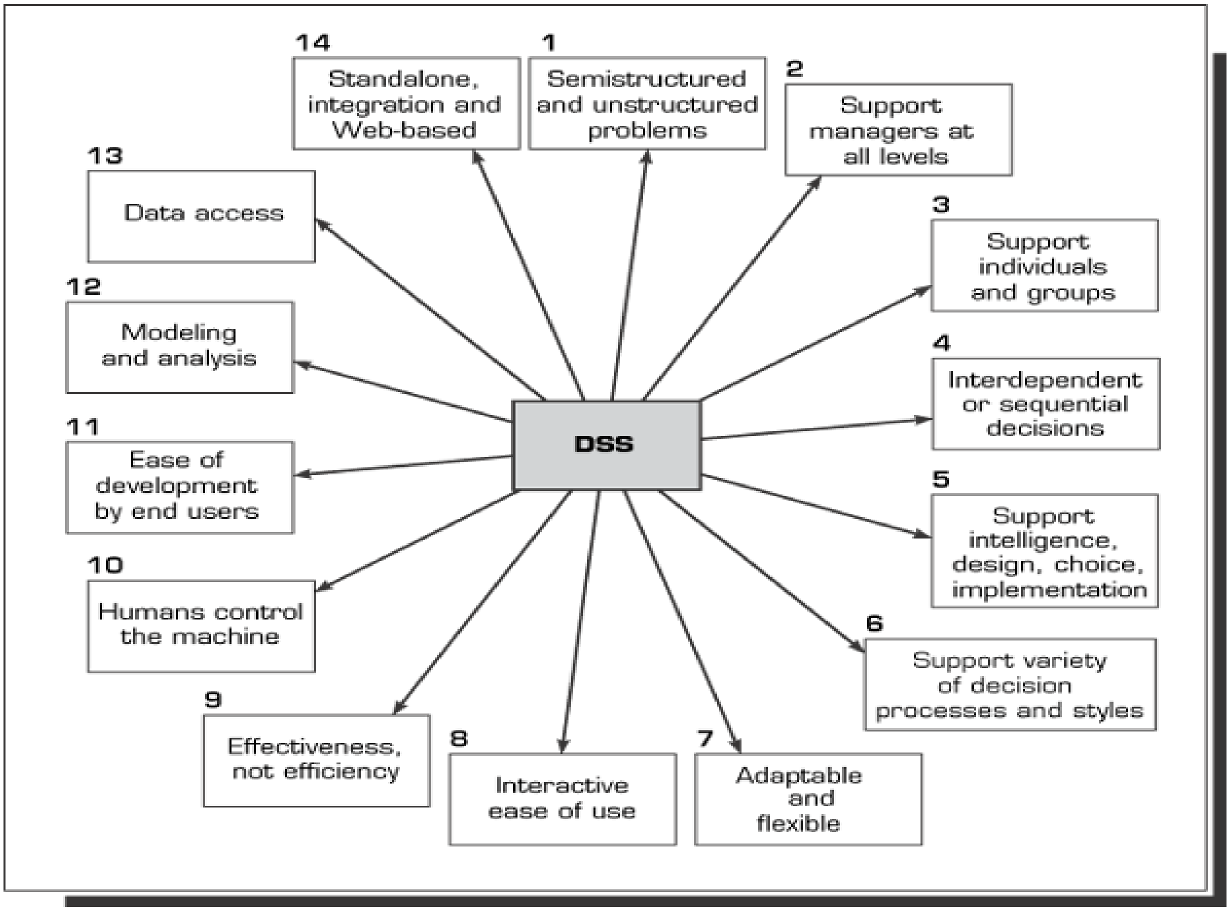
Decision Support Systems (DSS) are a class of computerized information system which designed to support managerial decision-making in unstructured problems and support decision-making activities. DSS are interactive computer-based systems and subsystems intended to help decision makers use communications technologies, data, documents, knowledge and/or models to complete decision process tasks. Typical information that a decision support application might gather and present would be,

- Accessing all information assets, including legacy and relational data sources;
- Comparative data figures;
- Projected figures based on new data or assumptions;
- Consequences of different decision alternatives, given past experience in a specific context.

A DSS is a methodology that supports decision-making. It is: Flexible; Adaptive; Interactive; GUI-based; Iterative; and Employs modeling.

3.2 Characteristics of DSS

Figure 3.2 Key Characteristics and Capabilities of DSS

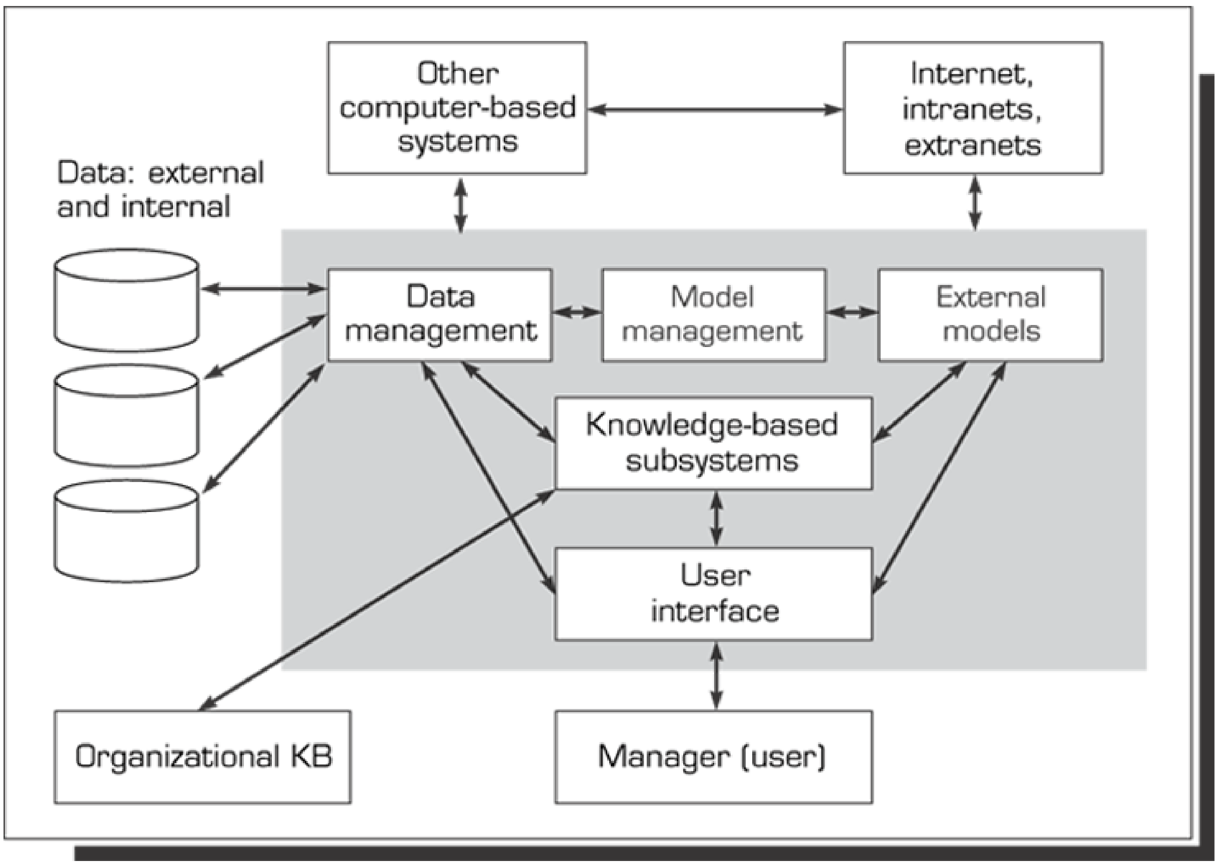


3.3 Components of DSS

- Three fundamental components of DSS:
  - the database management system (DBMS),
  - the model management system (MBMS), and
  - the dialog generation and management system (DGMS).
- the Data Management Component stores information (which can be further subdivided into that derived from an organization's traditional data repositories, from external sources such as the Internet, or from the personal insights and experiences of individual users);
- the Model Management Component handles representations of events, facts, or situations (using various kinds of models, two examples being optimization models and goal-seeking models); and

- User Interface Management Component is of course the component that allows a user to interact with the system

Figure 3.3 A Schematic View of DSS



3.4 What do Decision Support Systems Offer?

- Quick computations at a lower cost
- Group collaboration and communication
- Increased productivity
- Instant access to information stored in multiple databases and data warehouses
- Ability to analyze multiple alternatives and apply risk management
- Enterprise resource management
- Tools to obtain and maintain competitive advantage

The combination of the Internet, which enables speed and access, and the maturation of artificial intelligence techniques have led to sophisticated aids to support decision making under these risky and uncertain conditions. These aids have the potential to improve decision making by suggesting solutions that are better than those made by the human alone.They are increasingly available in diverse fields from medical diagnosis to traffic control to engineering applications.

3.5 Applications:

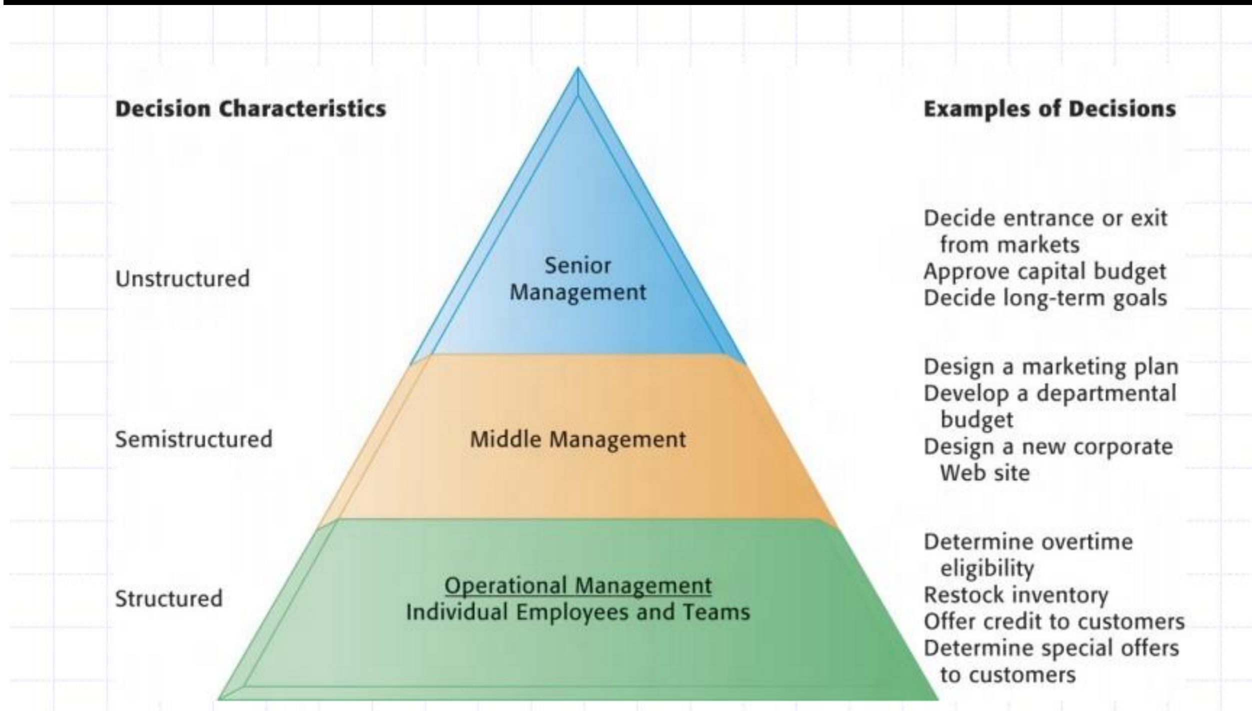
There are theoretical possibilities of building such systems in any knowledge domain.

- Clinical decision support system for medical diagnosis.
- a bank loan officer verifying the credit of a loan applicant
- an engineering firm that has bids on several projects and wants to know if they can be competitive with their costs.
- DSS is extensively used in business and management. Executive dashboards and other business performance software allow faster decision making, identification of negative trends, and better allocation of business resources.
- A growing area of DSS application, concepts, principles, and techniques is in agricultural production, marketing for sustainable development.
- A specific example concerns the Canadian National Railway system, which tests its equipment on a regular basis using a decision support system.
- A DSS can be designed to help make decisions on the stock market, or deciding which area or segment to market a product toward.

3.6 Types of Decision Problems

- Structured: situations where the procedures to follow when a decision is needed can be specified in advance
    - Repetitive
    - Standard solution methods exist
    - Complete automation may be feasible
    - Mostly use by operation level management
    -
- Unstructured: decision situations where it is not possible to specify in advance most of the decision procedures to follow
    - One-time
    - No standard solutions
    - Rely on judgment
    - Automation is usually infeasible
- Semi-structured: decision procedures that can be pre-specified, but not enough to lead to a definite recommended decision
    - Some elements and/or phases of decision making process have repetitive elements
    - DSSs most useful for repetitive aspects of semi-structured problems

TYPE OF DECISION:	TYPE OF CONTROL		
	Operational Control	Managerial Control	Strategic Planning
Structured (Programmed)	Accounts receivable, accounts payable, order entry	Budget analysis, short-term forecasting, personnel reports, make or buy	Financial management - Investments, warehouse locations, distribution centers
Semistructured	Production scheduling, inventory control	Credit evaluation, budget preparation, plant layout, project scheduling, rewards systems, inventory categorization	Building new plant, Mergers and acquisitions, new product planning, compensation, quality assurance, HR policy planning, inventory planning
Unstructured (Unprogrammed)	Selecting a cover for a magazine, Buying software, approving loans, help desk	Negotiations, recruiting an executive, hardware purchasing, lobbying	R&D planning, new technology development, social responsibility planning



3.7 Stages of decision making

- Define The Situation: Of the stages in decision making, this is probably the most significant step. It is vital to have a good understanding and be very clear about a) the situation and b) what you want to achieve.

- Generate alternatives :The number of alternatives you choose will depend on such factors as **experience, knowledge, skills**, number of people involved in generating alternatives and what's considered important.
    - Information gathering As alternatives are suggested, it may require further clarification of the situation or the decision to be made
    - Selection Selection is the choosing of one of the alternatives
    - Action Great decisions are only great when they are **carried into action** and the action achieves the desired result.
- Evaluate options that will solve the problem, pros, cons and risks of each alternative
- Select the best option- may be necessary to loop back and gather more info
- Develop a plan of action - and implement it!

4. Database for Decision Support System

Components of DSS

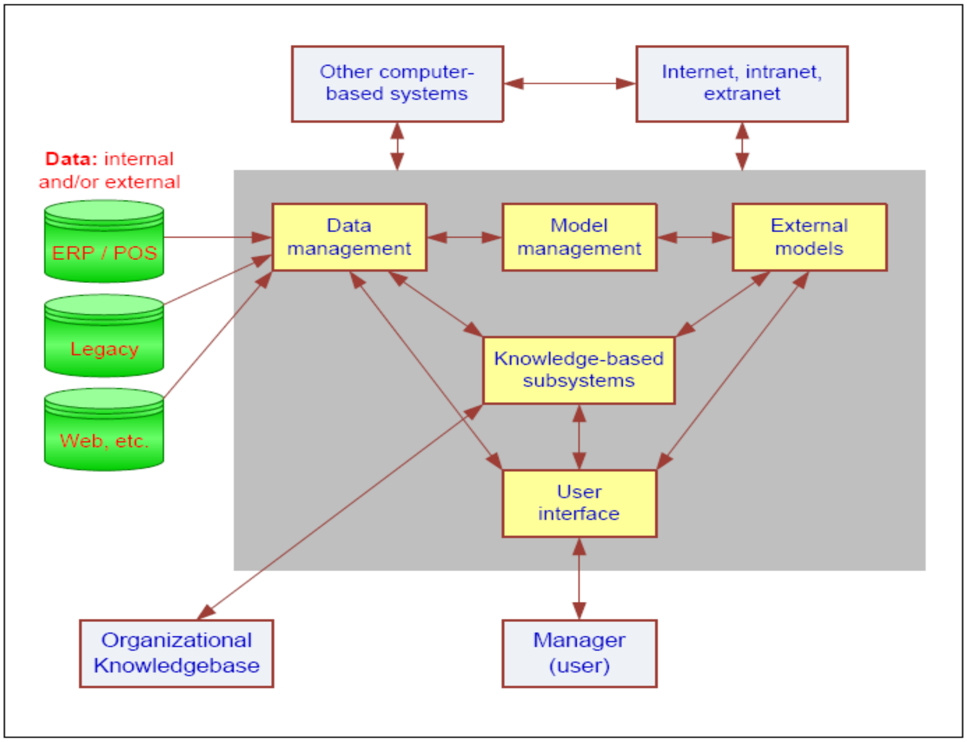
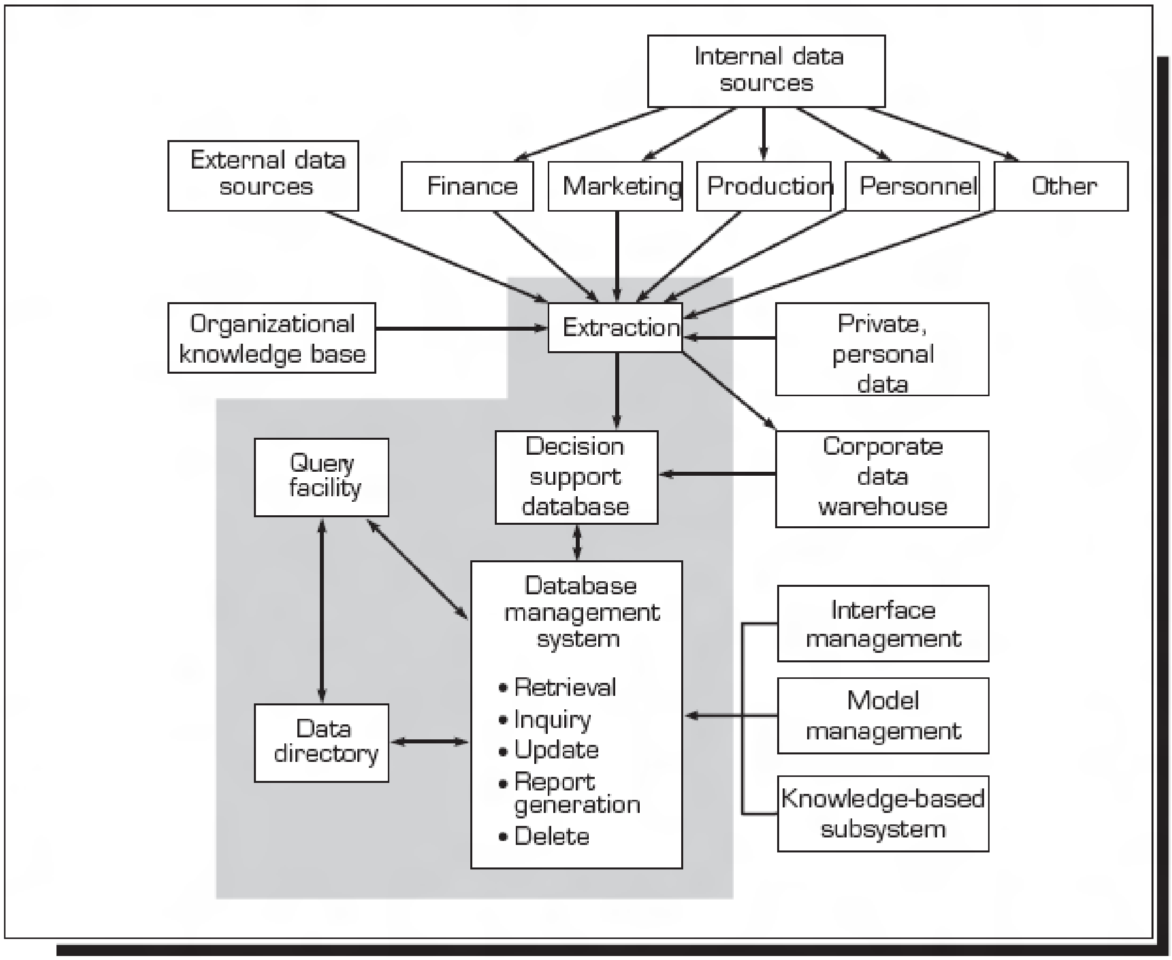


FIGURE 3.5 The Structure of the Data Management Subsystem



The Database :

- **Internal data come mainly from the organization's transaction processing system**
- **External data include industry data, market research data, census data, regional employment data, government regulations, tax rate schedules, and national economic data**
- **Private data can include guidelines used by specific decision makers and assessments of specific data and/or situations**

Data Organization

Data for DSS can be

- **entered directly into models**
- **extracted directly from larger databases e.g. Data Warehouse**

- **Can include multimedia objects**

Data extraction: ***The process of capturing data from several sources, synthesizing them, summarizing them, determining which of them are relevant,and organizing them, resulting in their effective integration.*** **The process of**

- **capturing data from several sources**
- **synthesizing, summarizing**
- **determining which of them are relevant**
- **and organizing them**
- **resulting in their effective integration**

A database is created, accessed and updated by a DBMS

- **Software for establishing, updating, and querying e.g. managing a database**
  - **record navigation**
  - **data relationships**
  - **report generation**

Query Facility

- **The (database) mechanism that**
  - **accepts requests for data**
  - **accesses**
  - **manipulates**
  - **and queries data**
- **Includes a query language**
  - **e.g. SQL**

Data Directory

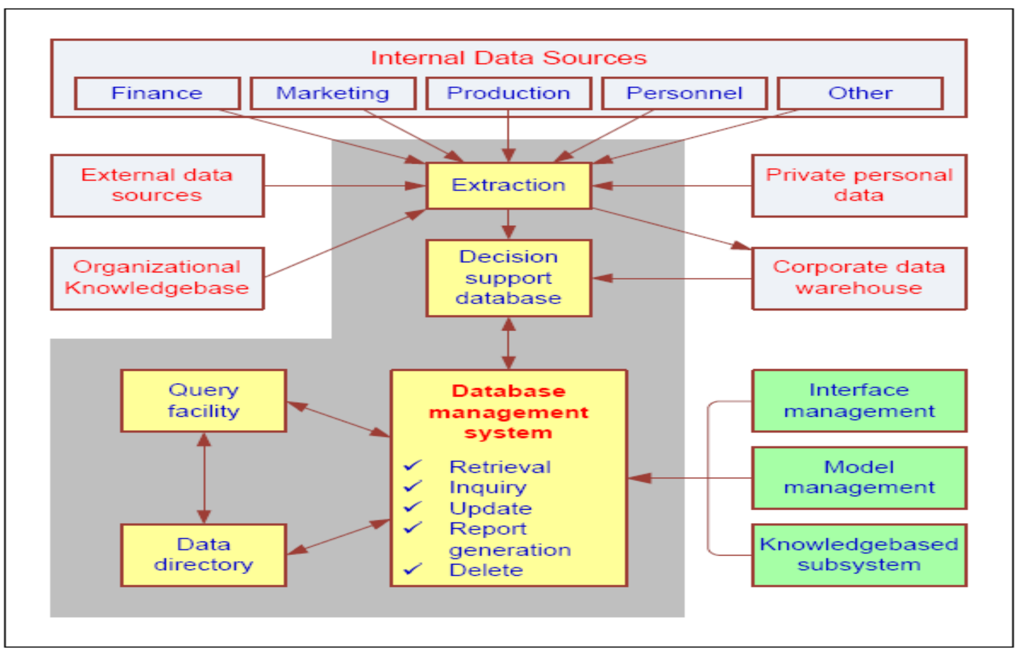
- **A catalog of all the data in a database or all the models in a model base**
- **Contains**
  - **data definitions**
  - **data source**
  - **data meaning**
- **Supports addition and deletion of new entries**

Key DB & DBMS Issues

- **Data quality**
  - **"Garbage in/garbage out" (GIGO)**

  - **Managers feel they do not get the data they need – 54% satisfied**
  - **Poor quality data leads to poor quality information**
    - **waste**
    - **lost opportunities**
    - **unhappy customers**
- **Data integration**
  - **For DSS to work, data must be integrated from disparate sources**
  - **"Creating a single version of the truth"**
- **Scalability**
  - **Volume of data increases dramatically**
    - **e.g. from 2001 – 2003, size of largest TPS DB increase two-fold (11 – 20 terabytes)**
  - **Needs new *storage* and *search* technologies**
- **Data security**
  - **data must be protected from unauthorized access through security measures**
  - **tools to monitor database activities**
  - **audit trail**

Database technologies can be applied into two types of scenarios:



- Transaction Processing(OLTP)

- Analytic Processing, using statistical method(OLAP) or machine/computational learning method(Data Mining)

OLTP, which is based on E.F. Codd's relation model, is the traditional (maybe most popular) application type of DBMS and most people are very familiar with it. This post tries to summarize related technologies in analytic processing, which is widely adopted in decision support systems.

Operational data (in OLTP system) is extracted, transformed(also cleaned) and loaded(by ETL subsystem) into the data warehouse for further analyzing. OLAP and DM systems read these data, analyze them, produce useful reports and present them to end(business) users.

OnLine Analytical Processing(OLAP)

OLAP is the processing of large scale multidimensional data using statistical based methods.

Multidimensional Model

Multidimensional model view data as as cubes that generalize spreadsheets to any number of dimensions. It categorizes data either as numerical values(a.k.a. measures) associated with some facts or textual values(a.k.a. dimensions) that characterize the facts.

OLAP Server Architecture

ROLAP (Relational OLAP)

Fact data is stored in relation model based storage system and some special induces technologies may be adopted. In this architecture, measures are derived from the records in the fact table and dimensions are derived from the dimension tables.from star schema example

This architecture can be further divided into two types:

Star Schema

Snowflake Schema

- MOLAP (Multidimensional OLAP)

Fact data is stored in an optimized multi-dimensional array storage, i.e., the server supports the multidimensional model directly.

- HOLAP (Hybrid OLAP)

It is a combination of ROLAP and MOLAP. HOLAP allows storing part of the data in a MOLAP store and another part of the data in a ROLAP store.

Unit -5

4. Modeling and DSS Model management

Decision Support Systems (DSS) are a class of computerized information system that support decision-making activities. DSS are interactive computer-based systems and subsystems intended to help decision makers use communications technologies, data, documents, knowledge and/or models to complete decision process tasks. Typical information that a decision support application might gather and present would be,

- Accessing all information assets, including legacy and relational data sources;
- Comparative data figures;
- Projected figures based on new data or assumptions;
- Consequences of different decision alternatives, given past experience in a specific context.

4.1 MODEL DRIVEN DSSs

Model-Driven DSS: This type of DSS emphasizes access to and manipulation of a model, e.g., statistical, financial, optimization and/or simulation. Simple statistical and analytical tools provide the most elementary level of functionality.

In general, model-driven DSS use complex financial, simulation, optimization and/or rule (expert) models to provide decision support. Model-driven DSS use data and parameters provided by decision makers to aid decision makers in analyzing a situation, but they are not usually data intensive, that is very large data bases are usually not need for model-driven DSS.

Simple examples: a spread-sheet with formulas, a statistical forecasting model. But also complex, initially statistical modelling oriented, software packages, like SAS or STATISTICA, as well as modules of multiple contemporary data manipulation systems of .... Further step: Business Analytics

4.2 DATA DRIVEN DSSs

This type of DSS emphasizes access to and manipulation of a time-series of internal company data and sometimes external data

- Basic level: queries and reports
- Intermediate level: Data Warehousing (Data Warehouse), increasing the efficiency of data sharing
- Highest level: use of advanced analytical techniques: On-line Analytical Processing (OLAP), and cross-sectional multilevel analysis (data cubes)

Most data-driven DSSs are targeted at managers, staff and also product/service suppliers. Data-driven DSS is a type of DSS that emphasizes access to and manipulation of a time-series of internal company data and sometimes external data. Simple file systems accessed by query and retrieval tools provide the most elementary level of functionality. Data warehouse systems that allow the manipulation of data by computerized tools tailored to a specific task and setting or by more general tools and operators provide additional functionality. Data-driven DSS with On-line Analytical Processing (OLAP) provides the highest level of functionality and decision support that is linked to analysis of large collections of historical data. Executive Information Systems (EIS) and Geographic Information Systems (GIS) are special purpose Data-Driven DSS.

- A Data Warehouse is a database designed to support decision making in organizations. It is batch updated and structured for rapid online queries and managerial summaries. Data warehouses contain large amounts of data. A data warehouse is a subject-oriented, integrated, time-variant, nonvolatile collection of data in support of management's decision making process.
- On-line Analytical Processing (OLAP) software is used for manipulating data from a variety of sources that has been stored in a static data warehouse. The software can create various views and representations of the data. For a software product to be considered an OLAP application it must contain three key features: 1. multidimensional views of data; 2. complex calculations; and 3. time oriented processing capabilities.
- Executive Information Systems (EIS) are computerized systems intended to provide current and appropriate information to support executive decision making for managers using a networked workstation. The emphasis is on graphical displays and an easy to use interface that present information from the corporate database. They are tools to provide canned reports or briefing books to top-level executives. EIS offer strong reporting and drill-down capabilities.
- A Geographic Information System (GIS) or Spatial DSS is a support system that represents data using maps. It helps people access, display and analyze data that have geographic content and meaning.

4.3 KNOWLEDGE DRIVEN DSSs

Knowledge-Driven DSS: can suggest or recommend actions to managers.

These DSS are person-computer systems with specialized problem-solving expertise. The "expertise" consists of knowledge about a particular domain, understanding of problems within that domain, and "skill" at solving some of these problems. The resulting systems are often called Expert Systems.

Tools used for building Knowledge-Driven DSS (knowledge acquisition, machine learning, uncertainty management) are sometimes called intelligent techniques, while the systems (tools) are called Intelligent Decision Support Systems. Such systems are close to the classical meaning of a DSS (interactive, limited use of data and models, suggest solution or recommendation).

Data Mining tools can be used to create Knowledge-driven DSS that have major data and knowledge components.

4.4 COMMUNICATIONS DRIVEN DSS

systems that use network and communications technologies to facilitate collaboration and communication. The communications technologies are central to supporting decision-making. Technologies include: LANs, WANs, Internet, ISDN, Virtual Private Networks.

4.5 Document-driven DSS - no need to introduce Google or Web of Science/Web of Knowledge. Such systems integrate a variety of storage and processing technologies to provide complete document retrieval and analysis. The Web provides access to large document databases including databases of hypertext documents, images, sounds and video. A search engine is a primary tool associated with a Document-Driven DSS.

Compression between data driven and model driven DSS:

Model-driven DSS	Data-driven DSS
User interacts primarily with a (mathematical) model and its results	User interacts primarily with the data
Helps to solve well-defined and structured problem (what-if-analysis)	Helps to solve mainly unstructured problems
Contains in general various and complex models	Contains in general simple models
Large amounts of data are not necessary	Large amounts of data are crucial
Helps to understand the impact of decisions on organizations	Helps to prepare decisions by showing developments in the past and by identifying relations or patterns

- What-if Analysis
  - End user makes changes to variables, or relationships among variables, and observes the resulting changes in the values of other variables
- Sensitivity Analysis
  - Value of only one variable is changed repeatedly and the resulting changes in other variables are observed
- Goal-Seeking
  - Reverse direction of analysis done in 'what-if' analysis
  - Set a target value for a variable and then repeatedly change other variables until the target value is achieved
  - Also called 'how can' analysis
- Optimization
  - More complex extension of goal-seeking analysis
  - Goal is to find the optimum value for one or more target variables given certain constraints
  - One or more other variables are changed repeatedly until the best values for the target variables are discovered (Microsoft Excel → Solver)

4.6 What if analysis:

In the early days of decision support systems, one of the major DSS "selling points" of vendors and academics was the ability to do "What If?" analysis. In the 1970s, model-driven DSS for sales and production planning allowed a manager to change a decision variable like the number of units to produce and then immediately get a new result for an outcome variable like profit. As DSS have gotten more sophisticated and become more diverse, the use of "What If?" as a concept has broadened.

In a nutshell, what-if analysis can be described as a data-intensive simulation whose goal is to inspect the behavior of a complex system (i.e., the enterprise business or a part of it) under some given hypotheses (called scenarios). More pragmatically, what-if analysis measures how changes in a set of independent variables impact on a set of dependent variables with reference to a given simulation model ; such model is a simplified representation of the business, tuned according to the historical enterprise data. A simple example of what-if query in the marketing domain is: How would my profits change if I run a 3 × 2 promotion for one week on some products on sale?

What-if analysis should not be confused with sensitivity analysis, aimed at evaluating how sensitive is the behavior of the system to a small change of one or more parameters. Besides, there is an important difference between what-if analysis and simple forecasting, widely used especially in the banking and insurance fields. In fact, while forecasting is normally carried out by extrapolating trends out of the historical series stored in information systems, what-if analysis requires to simulate complex phenomena whose effects cannot be simply determined as a projection of past data, which in turn requires to build a simulation model capable of reproducing – with satisfactory approximation – the real behavior of the business. For the same reason, the design of what-if applications is also more complex than that of conventional DWs, which only relies on a static model of business.

What if analysis - Observing how changes to selected variables affect other varibales. Example: What happens to sales if we cut advertising by 10%?

4.7 Sensitivity analysis (SA) is the study of how the uncertainty in the output of a model (numerical or otherwise) can be apportioned to different sources of uncertainty in the model input. A related practice is uncertainty analysis which focuses rather on quantifying uncertainty in model output. Ideally, uncertainty and sensitivity analysis should be run in tandem. Sensitivity analysis - Observing how repeated changes to a single variable affect other variables. Example: If advertising is cut by $100 repeatedly, to see how sales change

In more general terms uncertainty and sensitivity analysis investigate the robustness of a study when the study includes some form of statistical modeling. Sensitivity analysis can be useful to computer modelers for a range of purposes, including:

- Support decision making or the development of recommendations for decision makers (e.g. testing the robustness of a result);
- Enhancing communication from modelers to decision makers (e.g. by making recommendations more credible, understandable, compelling or persuasive);
- Increased understanding or quantification of the system (e.g. understanding relationships between input and output variables); and
- Model development (e.g. searching for errors in the model).

Let us give an example: in any budgeting process there are always variables that are uncertain. Future tax rates, interest rates, inflation rates, headcount, operating expenses and other variables may not be known with great precision. Sensitivity analysis answers the question, "if these variables deviate from expectations, what will the effect be (on the business, model, system, or whatever is being analyzed)?"

4.8 Goal-Seeking analysis: it compiles all of the given data and determines what inputs are required to reach specific goals. Sensitivity analysis is great and can be used to determine what portions of DSS, effect one and other. However, it does not look at the bottom line. It just demonstrates how portions interact with one and other. In addition, What-If analysis just looks as the possibilities and given scenarios. It attempts to determine how well things could or could not go. These are both great however, they fail to look at the overall picture. In order to reach goals, these specific requirements need to be met. In addition, What-if analysis uses Goal-seeking analysis. It looks at what numbers and goals are required in order to well, just average, and to do poorly. The whole purpose of the DSS is to compile raw data into useful information that managers can use effectively and apply to organizational and business decisions.

Goal-seeking analysis - Making repeated changes to selected variables until a chosen variable reaches a target value. Example, Increase advertising until sales reach $5,000,000 USD.

Optimization analysis - Finds an optimum value for selected variable given certain constraints: Example: What is the best amount of advertising to have, given budget and media choice?

Unit -6

6 DSS User Interface

6.1 Issue related to building an interface

An effective user interface is an important component of any type of Decision Support System, but it is especially important for systems that will be used directly by managers. In a Decision Support System, the user interface is sometimes called the dialogue or "front-end" component. Some might ask "why is the user interface or dialogue component of a DSS so important?" Research indicates that the easier it is to use a DSS interface, then the more "user friendly" most people will consider the system and hence the greater the chance that managers will actually use the DSS. The user interface is what managers see when they work with a DSS.

The goal of user interface design is developing screen layouts and interfaces that are easy to use and that are visually attractive (cf., Galitz, 1985). Both the intended users of a Decision Support System and information systems designers need to participate actively in designing and evaluating DSS user interfaces. Let's examine the concept of a user interface.

Most of these users do NOT want to learn a command language interface like Structured Query language (SQL) that may be used by an expert or by a more technically-oriented user. According to Bennett (1986), for a non-technical user the design of an appropriate DSS user interface is the most important determinant of the success of a decision support implementation.

A well-designed user interface can increase human processing speed, reduce errors, increase productivity and create a sense of user control. The quality of the system interface, from the user's perspective, depends upon what the user sees or senses, what the user must know to understand what is sensed, and what actions the user can and in some cases must take to obtain needed results.

Both groups need to be familiar with the following important issues and topics related to building and evaluating a user interface:

- User interface style – Is the style or combination of styles appropriate? What styles are used in the user interface?
- Screen design and layout -- Is the design easy to understand and attractive? Is the design symmetric and balanced?
- The Human-Software interaction sequence -- Is the interaction developed by the software logical and intuitive? Do people respond predictably to the interaction sequence?
- Use of colors, lines and graphics -- Are colors used appropriately? Do graphics improve the design or distract the user? 5. Information density -- Is too much information presented on a screen? Can users control the information density?
- Use of icons and symbols -- Are icons understandable?
- Choice of input and output devices -- Do devices fit the task?

Managers and analysts should focus on these seven design issues when they evaluate a DSS prototype or the proposed screens for a DSS.

6.2 Issue related to building an interface

The user interface determines how information is entered in a DSS and how it is displayed by a DSS. The interface also determines the ease and simplicity of learning and using the system. There are four general structures or interface styles that can be used to control interactions with computerized information systems. These styles are

- command-line interfaces,
- menu interfaces,
- point-and-click graphical interfaces, and
- question-and- answer interfaces.

Each style can be used in creating DSS user interfaces. The styles can often be combined usefully in a single application or set of related applications [see Galitz (1985); Shneiderman (1992); and Turban (1993)]. When building a user interface a designer should try to provide multiple ways to perform the same task. For example, a design may include a command-line interface, pull down menus for commands, and keyboard command equivalents. Many input devices including keyboard, mouse, touch pad, and voice inputs can be used to manipulate these four general interface styles.

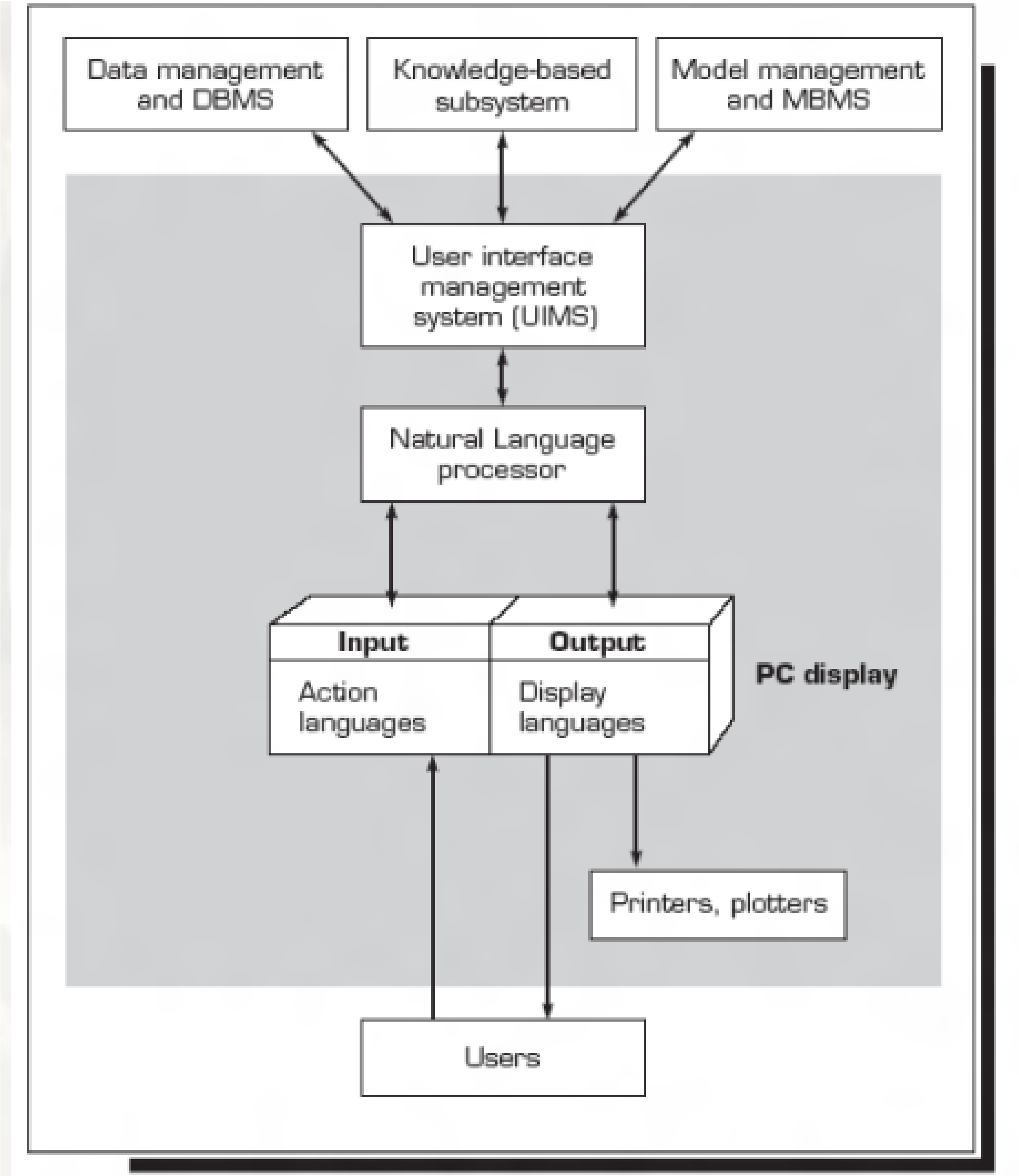


FIGURE 3.7 Schematic View of the User Interface System

Unit -7

7 Constructing a DSS :

7.1 DSS architecture

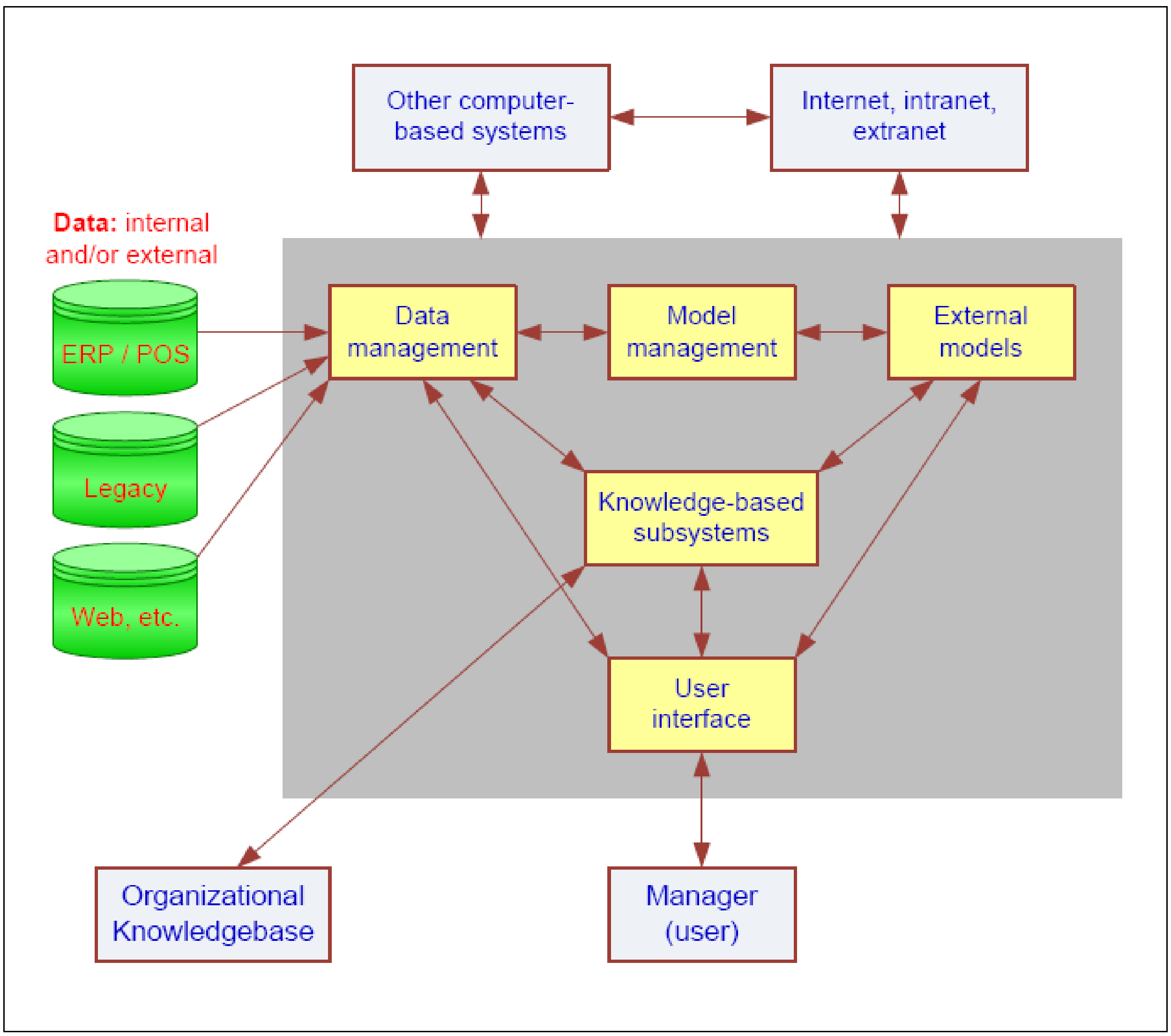
DSS for Management Support: DSS structure – DSS architecture [Turban 2011]

Many configurations exist, based on management-decision situation or specific technologies used for support

DSS have three **basic components**

1. Data
2. Model
3. User interface
4. (+ optional) Knowledge

Each component: has several variations; is typically deployed online, managed by a commercial of custom software



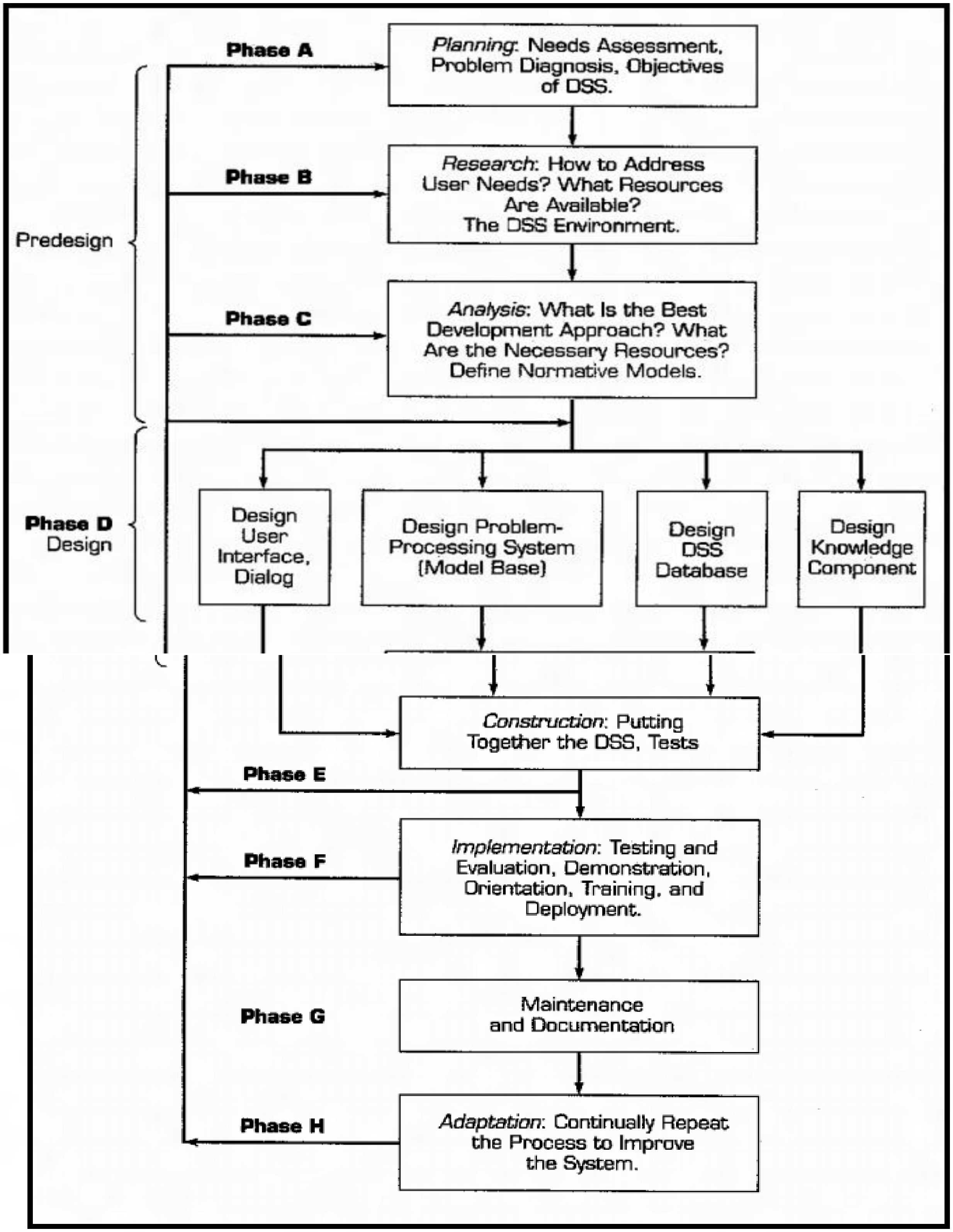
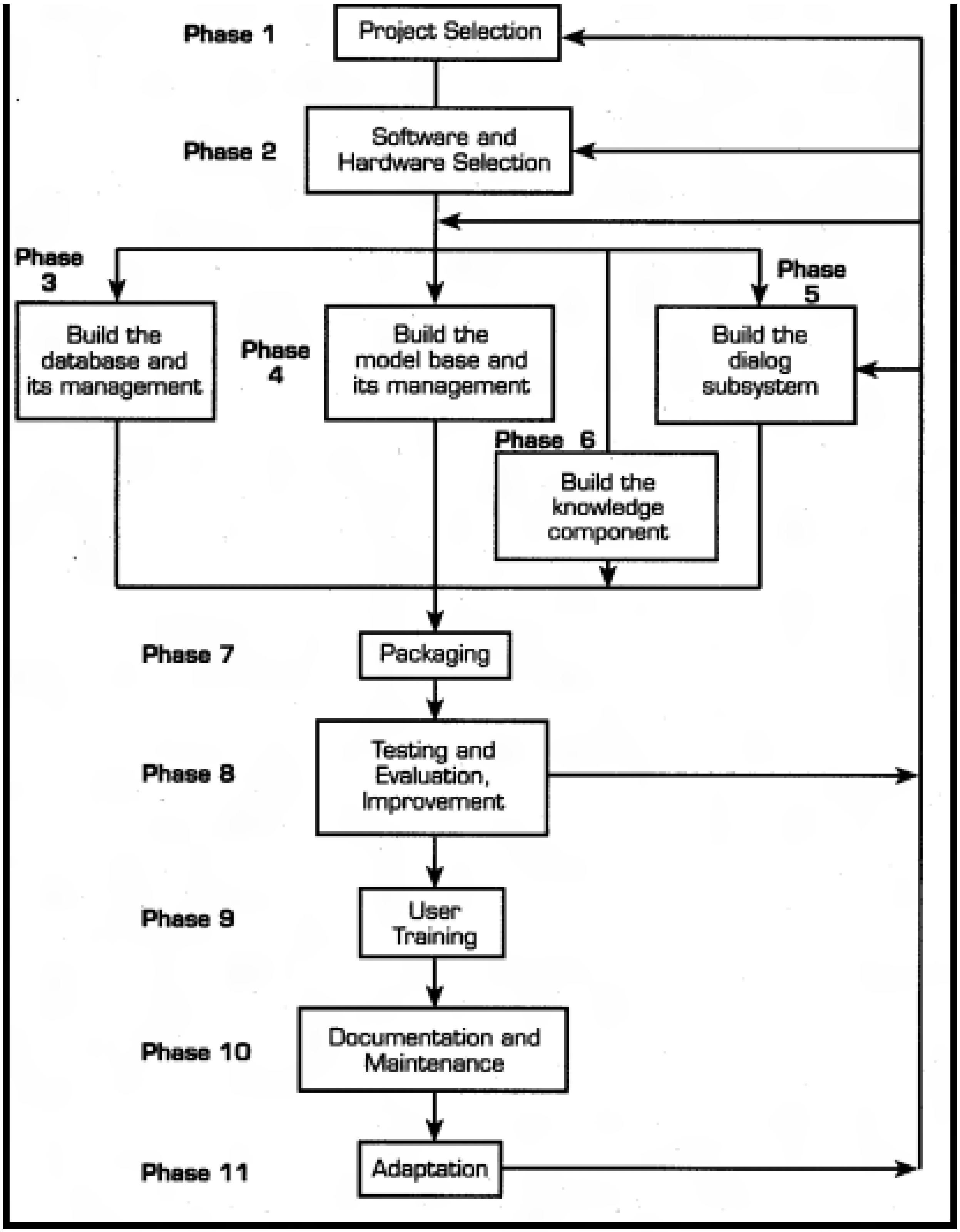


Figure 3-1 Phases in building a Decision Support System (Turban 1995)



e 3-2 The development process of a DSS constructed by end-users (Turban 1995)

Unit 8

8 Organizational DSS (ODSS) and Advanced Topics

8.1 Introduction and Characteristics

Organizational decision support systems (ODSS) are a class of decision support systems that promise to provide support at a higher organizational level for businesses than preceding forms of decision support. The existing literature provides many different descriptions of ODSS and its functionality. This leads to considerable confusion as to what is necessary for a system to be called an ODSS. This confusion has inhibited the development of ODSS both conceptually and in terms of implementation. We begin by considering the existing conceptual base upon which ODSS

The concept of organization decision support system (ODSS) is defined according to practical applications and novel understanding. And a framework for ODSS is designed. The framework has three components: infrastructure, decision-making process and decision execution process. Infrastructure is responsible to transfer data and information. Decision-making process is the ODSS's soul to support decision-making. Decision execution process is to evaluate and execute decision results derived from decision-making process. The framework presents a kind of logic architecture. An example is given to verify and analyze the framework

There is some controversy in the MIS literature concerning the potential impact of computer systems on interpersonal communication in organizations. Generally, MIS researchers have found that effective communication between users and designers is an important factor in determining user satisfaction and MIS success.

Recently, however, Naylor [8] has argued that decision support systems (DSS) may isolate managers from interaction with others. This is in marked contrast to the contentions of Wagner [16] and Huber [6] that DSS may lead to more effective managerial communication.

We would argue that DSS encourage communication for several reasons: adaptive DSS development requires continued manager-analyst interaction; DSS-based decisions often require groups of managers; and there is a greater need for human information processing in a DSS environment.

This paper reports the results of a study of the relationships between DSS usage and organizational communications. It was found that users in three different categories (managers, financial or planning analysts (FPAs), and “others”) felt that DSS usage encouraged communication. Also, significant positive relationships were found between increased communication and overall satisfaction with the DSS for all three groups. Finally, for managers and FPAs, significant positive relationships were found between increased communication and satisfaction with the DSS in decision-making activities.

Unit 9

9 Group Decision Support Systems (GDSS)

Group decision support systems (GDSS) are interactive, computer-based systems that facilitate solution of semi-structured and unstructured problems by a designated set of decision-makers working together as a group. A GDSS can assist groups, especially groups of managers, in analyzing problem situations and in performing group decision making tasks. GDSS include structured decision tools for tasks like brainstorming, commenting on ideas, and rating and ranking of alternatives (cf., DeSanctis and Gallupe, 1987).

Decision making is frequently a shared process. For example, meetings among groups of managers from different areas are an essential element for reaching consensus. The group may be involved in making a decision or in a decision-related task, like creating a short list of acceptable alternatives or deciding on criteria for accepting an alternative. When a decision-making group is supported electronically, the support is referred to as group decision support. Two types of groups are considered: a one-room group whose members are in one place (e.g., a meeting room), and a virtual group, whose members are in different location.

An increasing number of companies are useng GDSSs, especially when virtual groups are involved. One example is the Internal Revenue Service, which used a one-room GDSS to implement its quality-improvement programs based on the participation of a number of its quality teams. The GDSS was helpful in identifying problems, generating and evaluating ideas, and developing and implementing solutions. Another example is the European automobile industry, which used a one-room GDSS to examine the competitive automotive business environment and make ten-year forecasts, needed for strategic planning.

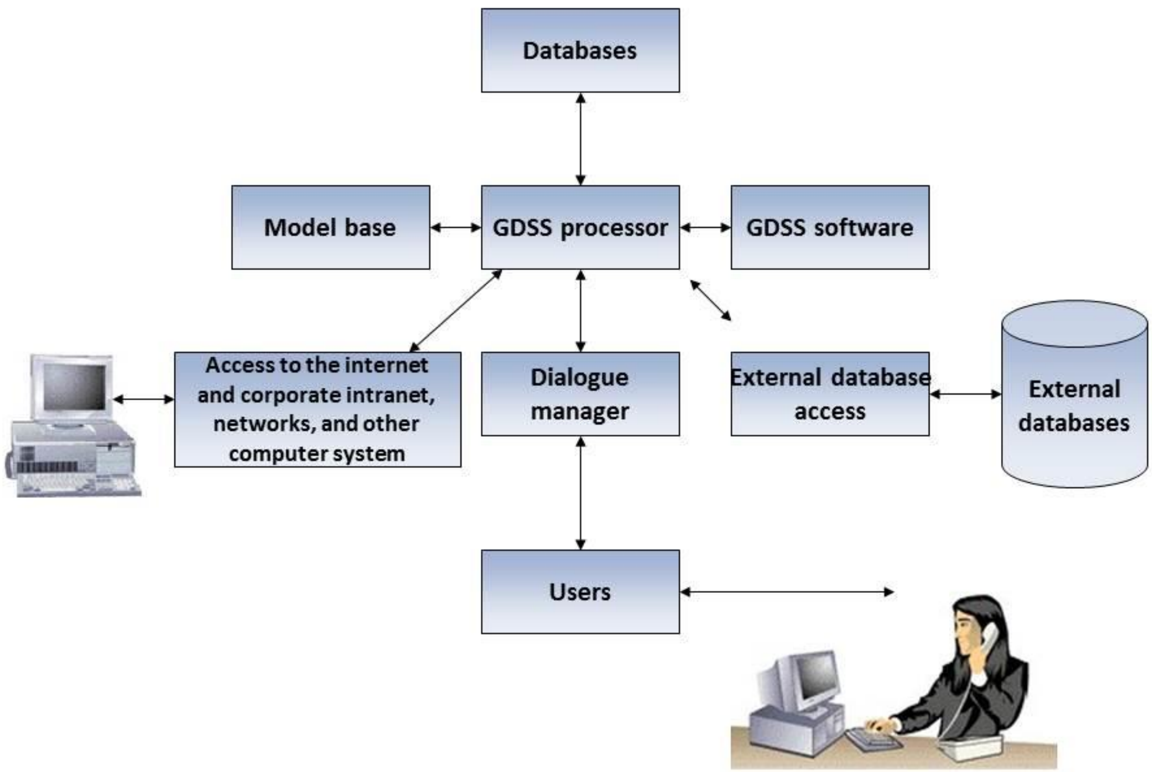
A group decision support system (GDSS) is an interactive computer-based system that facilitates the solution of semistructured and unstructured problems when made by a group of decision makers. The objective of a GDSS is to support the process of arriving at a decision. The first generation of GDSSs was designed to support face-to-face meetings in what is called decision room.

Group Decision Support Systems (GDSS) - An interactive, computer-based system that facilitates solution of unstructured problems by a set of decision-makers working together as a group. It aids groups, especially groups of managers, in analyzing problem situations and in performing group decision making tasks. Group Support Systems has come to mean computer software and hardware used to support group functions and processes.

Characteristics of Group Decision Support System

- Designed with the goal of supporting groups of decision-makers in their work.
- The GDSS accommodates users with varying levels of knowledge regarding computing and decision support.
- Can be designed for one type of problem or for a variety of group-level organizational decisions.
- Encourages generation of ideas, resolution of conflicts, and freedom of expression.

- Supporting parallel processing of information and idea generation by participants.
- Enabling larger groups with more complete information, knowledge, and skills to participate in the same meeting.
- Permitting the group to use structured or unstructured techniques and methods of arriving at decisions.
- Offering rapid and easy access to external information.
- Producing instant, anonymous voting results.
- Enabling several users to interact simultaneously.
- Automatically recording all information that passes through the system for future analysis (it



The Advantage, or process gains, from using a GDSS (over more traditional group techniques) are:

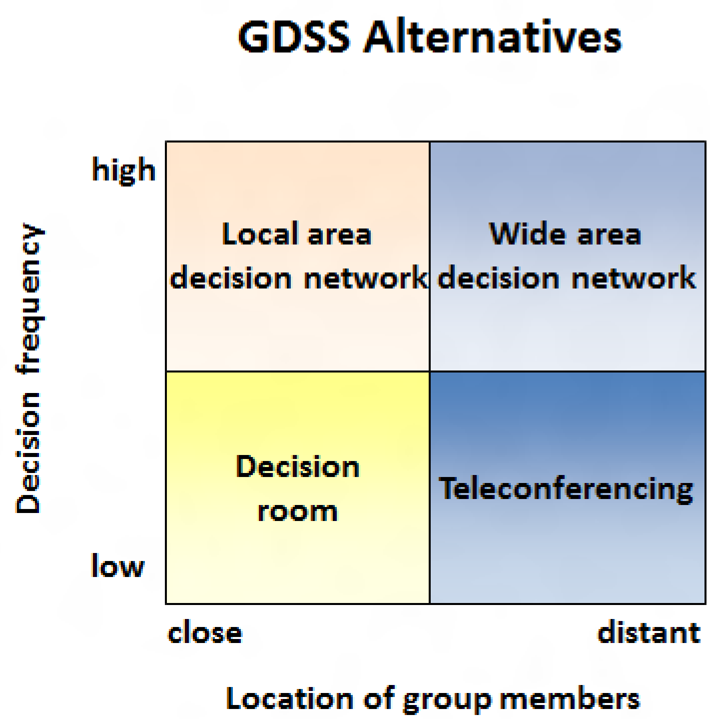
- More precise communication;
- Synergy: members are empowered to build on ideas of others;
- More objective evaluation of ideas;
- Stimulation of individuals to increase participation;
- More participation
- Group synergy

- Automated record keeping

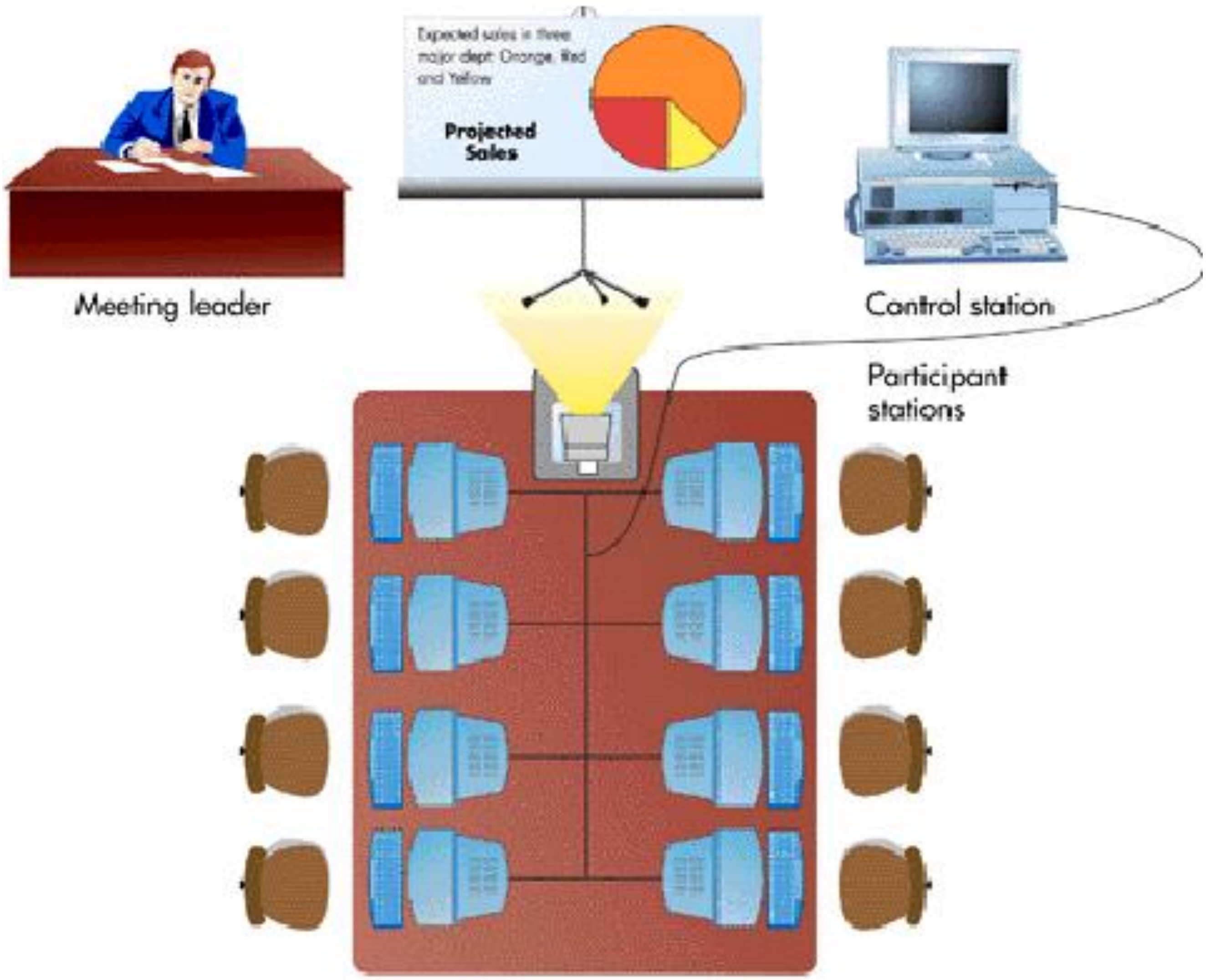
Disadvantages

There are some disadvantages to the technology, however, and they include:

- Slow Communication
- Not all Tasks are Amenable to GDSSs



- **Decision Room**
  - For decision makers located in the same geographic area or building
  - Use of computing devices, special software, networking capabilities, display equipment, and a session leader
  - Collect, coordinate, and feed back organized information to help a group make a decision
  - Combines face-to-face verbal interaction with technology-aided formalization



The Decision Room refers to the physical arrangement where the use of decision room software or Group DSS is normally utilized:

- For decision makers located in the same geographic area or building
- With the use of computing devices, special software, networking capabilities, display equipment and a session leader
- To collect, coordinate and feedback organized information to help a group make a decision and
- Combines face-to-face verbal interaction with technology-aided formalization.

The objective in using a Decision Room is to enhance and improve the group's decision-making process. Characteristics of a Decision Room include:

- Each participant has a computer workstation or terminal for anonymous input;
- A leader (facilitator) coordinates the meeting;

- The room has a display screen that all participants can view;
- Computers are networked and client/server architecture is used; and
- Specialized software is available to all participants for "brainstorming" and to support voting.

Wide Area Decision Network

- Characteristics
  - Location of group members is distant
  - Decision frequency is high
  - Virtual workgroups
    - Groups of workers located around the world working on common problems via a GDSS

The Time/Place Matrix

	Same place	*Different place*
Same time	* Decision room * Presentation software * PC screen sharing software	* Teleconferencing * Video * Audio * Personal
Different time	* Group memory management software * Kiosks * Group displays	* E-mail * Workflow software * Group authoring software * Computer conferencing

TYPES OF GDSS:

- connection management systems
  - providing a physical mechanism through which people involved in a decision can communicate
  - e.g.: WAN architecture
- communication management:
  - enhance information flow by means of facilities to store, reply, forward etc
  - e.g. electronic mail packages and discussion groups
- content management systems

  - provides intelligent routing – the system knows where a document goes after its current user finishes with it, or where the messages should go once it is entered
  - e.g. decision conference systems
- process management
  - considering the content of the information in the flow in deciding what to do with it

Levels of Group Decision Support Systems

There are three levels of Group Decision Support Systems:

- Level 1 GDSS - provide technical features aimed at removing common communication barriers such as voting, electronic message exchange
- Level 2 GDSS - provide decision modeling and group decision techniques aimed at reducing uncertainty that occur in the group's decision process.
- Level 3 GDSS - machine-induced group communication patterns and can include expert advice in the selecting and arranging of rules to be applied during a meeting.

Unit- 10

10. Distributed Group Support Systems

Distributed Group Support Systems use asynchronous computer mediated communication to support anytime/anywhere group discussions and decision making. GDSS type tools generally appear to improve both objective and subjective outcomes, various process interventions have had little or no effect on these groups, which had one to four weeks to adapt the use of the system features to their own expectations and preferences.

One type of computer-based system to support collaborative work is most often called a Group Support System, or GSS. Other terms that have been used include "Group Decision Support Systems" ("GDSS;" and "Electronic Meeting Systems.

Benefit of Group work

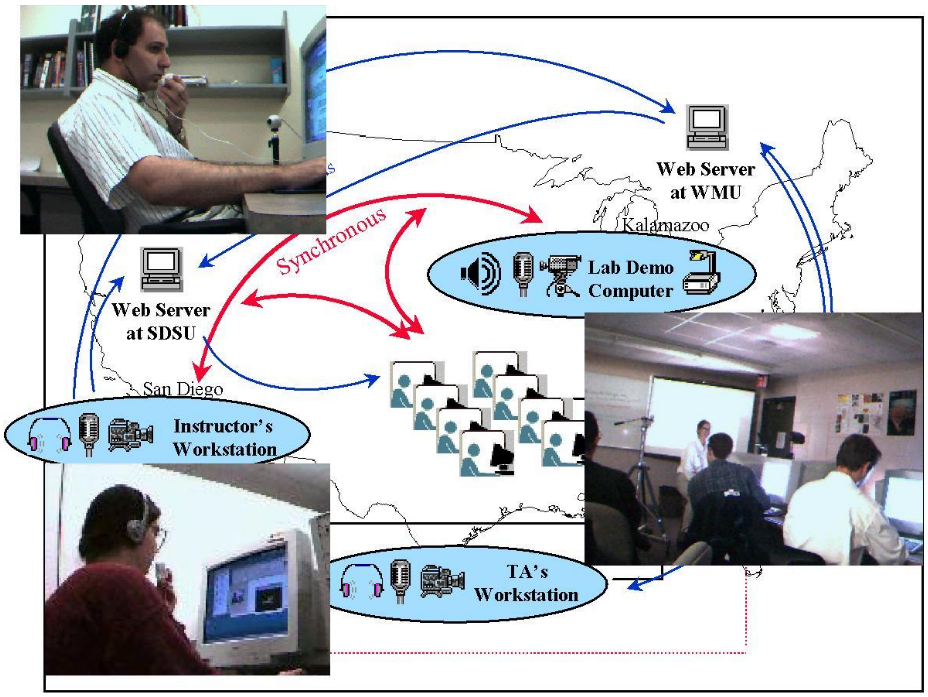
- To facilitate communication: faster, clearer, more persuasive
- Telecommuting
- Cut down on travel costs
- Bring together multiple perspectives and expertise
- Knowledge management
- Form groups with common interests where it wouldn't be possible to gather a sufficient number of people face-to-face
- Save time and cost in coordinating group work
- Facilitate group problem-solving
- Enable new modes of communication, such as anonymous interchanges or structured interactions
- Saves meeting time
- Leveraging professional expertise

Problems of Distributed support system

- Significantly more difficult to get right than traditional software
- System can't succeed unless most or all the target group is willing to adopt the system
- Integration into existing systems can be difficult
- Testing is difficult and roles change due to volatility of groups
- Resistance to change
- Problems are not technical, but social

Distance learning is a process of transferring knowledge to learners (students) who are separated from the instructor (teacher) by time and/or physical distance and are making use of technology components, such as the Internet, video, CD's, tapes, and other forms of technology to accomplish learning.

Is there any difference between "distance learning" and "distance education"? These two terms are used synonymously in education and learning technologies. It is believed that distance learning has been a method of teaching and learning for many individuals for at least one hundred years (Moore & Kearsley, 1996) starting with print technology and the postal service (correspondence education) all the way up to the electronic communication that is used today.



Distance Learning (DL) is an instructional delivery system that connects learners with educational resources. DL provides educational access to learners not enrolled in educational institutions and can augment the learning opportunities of current students. The implementation of DL is a process that uses available resources and will evolve to incorporate emerging technologies

There are two distance education delivery system categories - synchronous and asynchronous. Synchronous instruction requires the simultaneous participation of all students and instructors. The advantage of synchronous instruction is that interaction is done in "real time" and has an immediacy. Examples include interactive tele-courses, teleconferencing and web conferencing, and Internet chats.

Asynchronous instruction does not require the simultaneous participation of all students and instructors. Students do not need to be gathered together in the same location at the same time. Rather, students may choose their own instructional time frame and interact with the learning materials and instructor according to their schedules. Asynchronous instruction is more flexible than synchronous instruction, but experience shows that time limits are necessary to maintain focus and participation. The self-paced format accommodates multiple learning levels and schedules. Examples of asynchronous delivery include e-mail, audiocassette courses, videotaped courses, correspondence courses, and WWW-based courses.

Computer conferencing occurs when two or more users are teleconferencing using two or more computers. Usually, one of the conference participants is in charge of handling what is on the computer screen for all others to see. That person is referred to as a host or moderator. Individuals can all participate in a conference call while looking at one presentation on the computer. They can also communicate by instant messaging. Some computer conferences occur simultaneously with a telephone conference. That way, participants can chat via telephone while they watch what's occurring on the screen.

They only need an Internet connection. A host sends out an email to participants with a link to the conference room or web room and the date and time of the conference. If someone plans to participate, he can accept the invitation right from the email

There are several features of computer conferencing. First, most participants have the ability to set their notifications if they're going to be busy, on the phone, or away from their computer screen during the conference. The instant message feature allows participants to ask questions. The questions may go to all of the participants or can simply go to the host of the conference. Participants can also download manuals or notes in pdf format to accompany the conference. Quizzes and exercises can be taken during breaks in the conference. This is perfect for eLearning, as students have hands-on activities they can perform that will help them retain what they learn. Electronic bulletin boards can be set up for information or questions to be posted.

A misconception of computer conferencing is that people may be under the impression that they'll receive answers immediately. This is not always the case. Often, websites and forums are not moderated 100 percent of the time. You may post a question but not receive an answer quickly, or even at all. Follow-ups may be necessary.

Unit- 11

11. Executive Information and Support Systems

An Executive Support System ("ESS") is designed to help senior management make strategic decisions. It gathers, analyses and summarises the key internal and external information used in the business.

An executive support system also referred to as ESS is a smart and useful reporting tool that turns your organization's data into summarized reports.

Reports are usually needed by executive managers in order to have quick access to company details at all levels and also information on departments such as billing, staffing, scheduling and others. Executive support systems are also used for making analysis in order to predict performance outcomes and reports. Executives mainly use ESS to quickly see statistics and numbers that are needed for decision-making.

An Executive Information Systems (EIS) is a type of management information system intended to facilitate and support the information and decision making needs of senior executives by providing easy access to both internal and external information relevant to meeting the strategic goals of the organization. It is commonly considered as a specialized form of a Decision Support System (DSS) and otherwise referred to as an Executive Support System (ESS).

In recent years and in the USA, the term EIS has lost popularity in and the terms "business intelligence" and "online analytical processing" are often used for these types of applications.

Types of Executive Information System

- Corporate Management - responsible for business and fiscal planning, budgetary control, as well as for ensuring the corporate information technology needs are met in a co-ordinated and cost effective manner. E.g., Management functions, human resources, financial data, correspondence, performance measures, etc. (whatever is interesting to executives)
- Technical Information Dissemination – for the purpose of disseminating the latest information on relevant technologies, products, processes and markets E.g., Energy, environment, aerospace, weather, etc.

Executive Support Systems Characteristics.

A number of definitions have been put forward to describe EISs. While a definition is useful, in a complex area such as EISs a better understanding is obtained by looking at their characteristics. Some of these are given below:

- Executive support systems are end-user computerised information systems operated directly by executive managers. They utilise newer computer technology in the form of data sources, hardware and programs, to place data in a common format, and provide fast and easy access to information.
- They integrate data from a variety of sources both internal and external to the organisation.
- They focus on helping executives assimilate information quickly to identify problems and opportunities. In other words, EISs help executives track their critical success factors.

Capabilities of Executive Support Systems

Most executive support systems offer the following capabilities:

- Consolidation – involves the aggregation of information and features simple roll-ups to complex groupings of interrelated information
- Drill-down – enables users to get details, and details of details, of information
- Slice-and-dice – looks at information from different perspectives
- Digital dashboard – integrates information from multiple components and presents it in a unified display.

Advantages of Executive Information System

- As more executives come up through the ranks, they are more familiar with and rely more on technology to assist them with their jobs. Executive Support Systems don't provide executives with ready- made decisions. They provide the information that helps them make their decisions. Executives use that information, along with their experience, knowledge, education, and understanding of the corporation and the business environment as a whole, to make their decisions.
- Executives are more inclined to want summarized data rather than detailed data (even though the details must be available). ESS rely on graphic presentation of information because it's a much quicker way for busy executives to grasp summarized information
- It provides timely delivery of company summary information.
- It provides better understanding of information
- It filters data for management.
- It provides system for improvement in information tracking
- It offers efficiency to decision makers.

Disadvantages of Executive Information System

- Functions are limited, cannot perform complex calculations.
- Hard to quantify benefits and to justify implementation of an EIS.
- Executives may encounter information overload.
- System may become slow, large, and hard to manage.
- Difficult to keep current data.
- May lead to less reliable and insecure data.
- Small companies may encounter excessive costs for implementation.
- Highly skilled personnel requirement can not be fulfilled by the small business.

Unit -12

12. Overview of Applied Artificial Intelligence (AI) and Problem Solving

12.1 AI concepts ,characteristics

Intelligence is the ability to think, to imagine, creating, memorizing, and understanding, recognizing patterns, making choices, adapting to change and learn from experience. Artificial intelligence is a human endeavor to create a non-organic machine-based entity that has all the above abilities of natural organic intelligence

A scientific and engineering discipline devoted to: understanding principles that make intelligent behavior possible in natural or artificial systems; developing methods for the design and implementation of useful, intelligent artifacts.

A field of study that explores how computers can be used for tasks that require human characteristics of: intelligence, imagination , intuit ion

Operational Definition of AI

"The exciting new effort to make computers think ... *machines with minds*, in the full and literal sense" (Haugeland, 1985) "[The automation of] activities that we associate with human thinking, activities such as decision-making, problem solving, learning ..." (Bellman, 1978)	"The study of mental faculties through the use of computational models" (Charniak and McDermott, 1985) "The study of the computations that make it possible to perceive, reason, and act" (Winston, 1992)
"The art of creating machines that perform functions that require intelligence when performed by people" (Kurzweil, 1990) "The study of how to make computers do things at which, at the moment, people are better" (Rich and Knight, 1991)	"A field of study that seeks to explain and emulate intelligent behavior in terms of computational processes" (Schalkoff, 1990) "The branch of computer science that is concerned with the automation of intelligent behavior" (Luger and Stubblefield, 1993)

Systems that *think* like humans	Systems that *think* rationally
Systems that *act* like humans	Systems that *act* rationally

In order for something to be considered an "Artificial Intelligence," there are a few different characteristics that are required... Some of these characteristics include the following abilities:

- The ability to act intelligently, as a human.
- The ability to behave following "general intelligent action."
- The ability to artificially simulate the human brain.

- The ability to actively learn and adapt as a human.
- The ability to process language and symbols.

As can be seen from just these few examples, Artificial Intelligence primarily concerns the ability for a computer to mimic human intelligence. That is its key characteristic.

12.2 Practical Uses and Applications

- **Game playing** You can buy machines that can play master level chess for a few hundred dollars. There is some AI in them, but they play well against people mainly through brute force computation--looking at hundreds of thousands of positions. To beat a world champion by brute force and known reliable heuristics requires being able to look at 200 million positions per second.
- **speech recognition** In the 1990s, computer speech recognition reached a practical level for limited purposes. Thus United Airlines has replaced its keyboard tree for flight information by a system using speech recognition of flight numbers and city names. It is quite convenient. On the the other hand, while it is possible to instruct some computers using speech, most users have gone back to the keyboard and the mouse as still more convenient.
- **Understanding natural language** Just getting a sequence of words into a computer is not enough. Parsing sentences is not enough either. The computer has to be provided with an understanding of the domain the text is about, and this is presently possible only for very limited domains.
- **computer vision** The world is composed of three-dimensional objects, but the inputs to the human eye and computers' TV cameras are two dimensional. Some useful programs can work solely in two dimensions, but full computer vision requires partial three-dimensional information that is not just a set of two-dimensional views. At present there are only limited ways of representing three-dimensional information directly, and they are not as good as what humans evidently use.
- **expert systems** A ``knowledge engineer'' interviews experts in a certain domain and tries to embody their knowledge in a computer program for carrying out some task. How well this works depends on whether the intellectual mechanisms required for the task are within the present state of AI. When this turned out not to be so, there were many disappointing results. One of the first expert systems was MYCIN in 1974, which diagnosed bacterial infections of the blood and suggested treatments. It did better than medical students or practicing doctors, provided its limitations were observed. Namely, its ontology included bacteria, symptoms, and treatments and did not include patients, doctors, hospitals, death, recovery, and events occurring in

time. Its interactions depended on a single patient being considered. Since the experts consulted by the knowledge engineers knew about patients, doctors, death, recovery, etc., it is clear that the knowledge engineers forced what the experts told them into a predetermined framework. In the present state of AI, this has to be true. The usefulness of current expert systems depends on their users having common sense.

- **heuristic classification** One of the most feasible kinds of expert system given the present knowledge of AI is to put some information in one of a fixed set of categories using several sources of information. An example is advising whether to accept a proposed credit card purchase. Information is available about the owner of the credit card, his record of payment and also about the item he is buying and about the establishment from which he is buying it (e.g., about whether there have been previous credit card frauds at this establishment).

4.1 DSS and Problem Solving

Unit 13

13. Fundamentals of Expert Systems

An expert system is a computer system that emulates, or acts in all respects, with the decision-making capabilities of a human expert. An expert system is functionally equivalent to a human expert in a specific problem domain of reasonable complexity. The equivalence in qualified in terms of its capability to:

- Reason over representations of human knowledge
- Solve the problem by heuristic or approximation techniques
- Explain and justify the solution based on known facts

- Definitions of an Expert System
    - **A computer system whose performance is guided by specific, expert knowledge in solving problems.**
    - **A computer system that simulates the decision- making process of a human expert in a specific domain.**
    - **One of the early (large- scale) successes of artificial intelligence.**
    - **An expert system is an "intelligent" program that solves problems in a narrow problem area by using high-quality, specific knowledge rather than an algorithm.**

So why do we need to create an Expert System if a human expert can solve the same problem? Well here are a few reasons :

- Low cost of expertise
- Response without cognitive bias or logical fallacies
- 24/7 availability with high reliability
- Clone the expert
- Use in hazardous environments
- Ability to explain the reasoning without attitude.

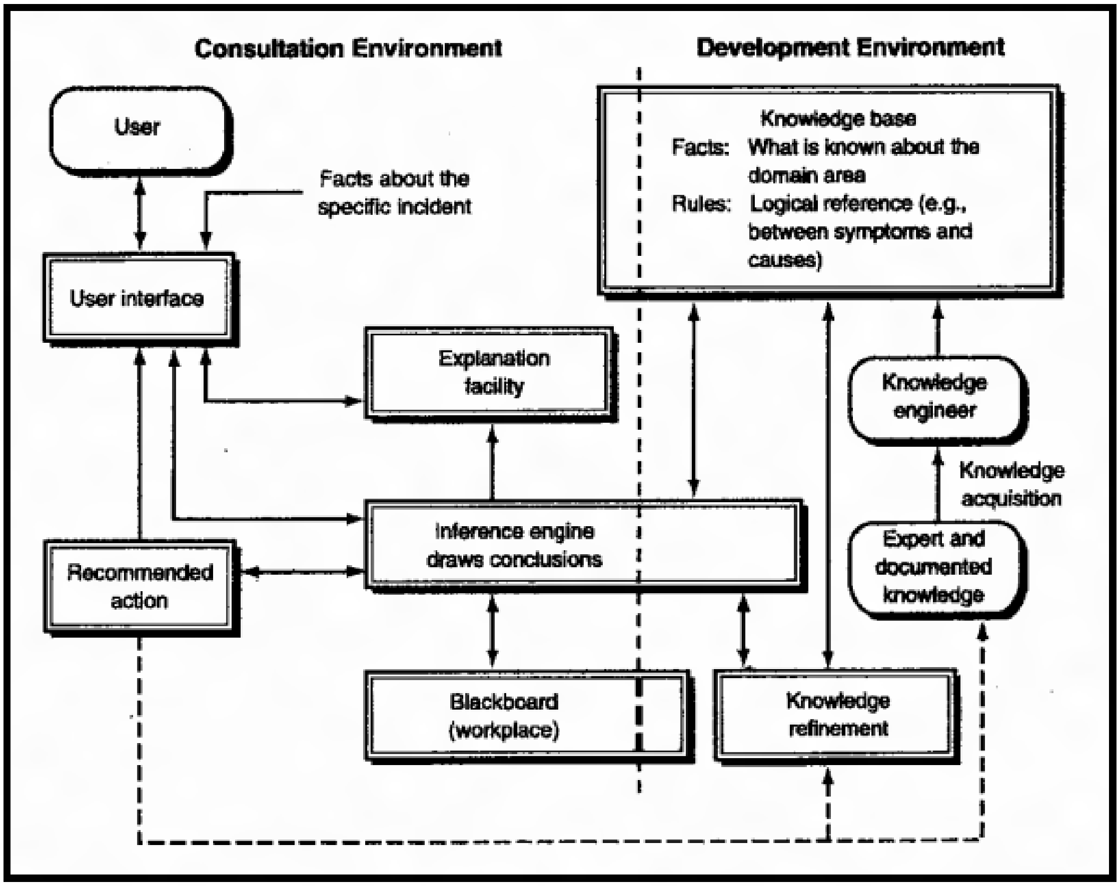
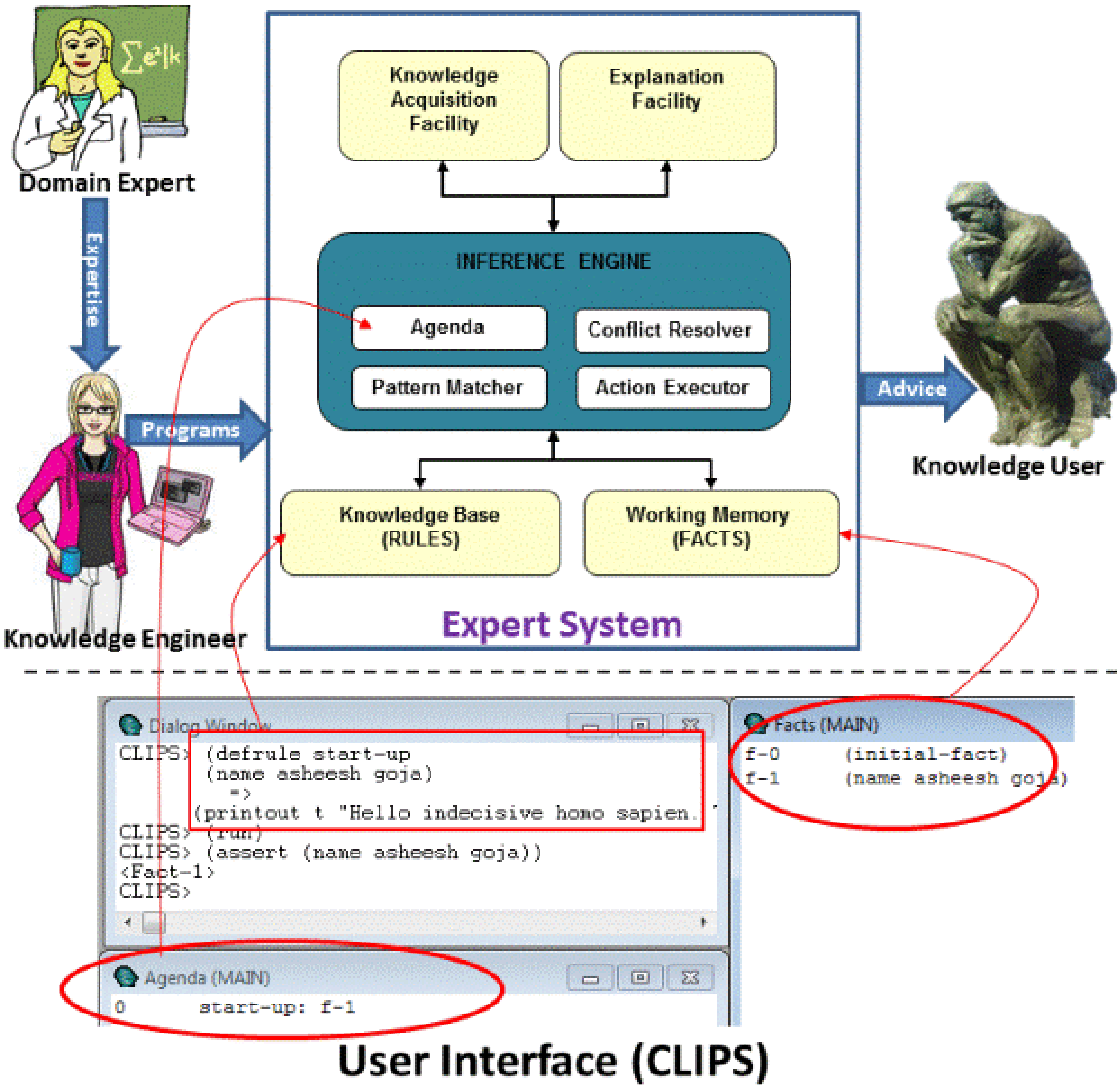


Figure 2-6 Structure and process of an expert system (Turban et al 2001)


Domain Expert
Expertise
Knowledge Engineer
Programs
Knowledge Acquisition Facility
Explanation Facility
INFERENCE ENGINE
Agenda
Conflict Resolver
Pattern Matcher
Action Executor
Knowledge Base (RULES)
Working Memory (FACTS)
Expert System
Advice
Knowledge User
Dialog Window
CLIPS> (defrule start-up
(name asheesh goja)
=>
(printout t "Hello indecisive homo sapien.
CLIPS> (run)
CLIPS> (assert (name asheesh goja))
<Fact-1>
CLIPS>
Facts (MAIN)
f-0 (initial-fact)
f-1 (name asheesh goja)
Agenda (MAIN)
0 start-up: f-1
User Interface (CLIPS)

Applications of expert systems

Applications tend to cluster into seven major classes.

Diagnosis and Troubleshooting of Devices and Systems of All Kinds

This class comprises systems that deduce faults and suggest corrective actions for a malfunctioning device or process. Medical diagnosis was one of the first knowledge areas to which ES technology was applied (for example, see Shortliffe 1976), but diagnosis of engineered systems quickly surpassed medical diagnosis. There are probably more diagnostic applications of ES than any other type. The diagnostic problem can be stated in the abstract as: given the evidence presenting itself, what is the underlying problem/reason/cause?

Planning and Scheduling

Systems that fall into this class analyze a set of one or more potentially complex and interacting goals in order to determine a set of actions to achieve those goals, and/or provide a detailed temporal ordering of those actions, taking into account personnel, materiel, and other constraints. This class has great commercial potential, which has been recognized. Examples involve airline scheduling of flights, personnel, and gates; manufacturing job-shop scheduling; and manufacturing process planning.

Configuration of Manufactured Objects from Subassemblies

Configuration, whereby a solution to a problem is synthesized from a given set of elements related by a set of constraints, is historically one of the most important of expert system applications. Configuration applications were pioneered by computer companies as a means of facilitating the manufacture of semi-custom minicomputers (McDermott 1981). The technique has found its way into use in many different industries, for example, modular home building, manufacturing, and other problems involving complex engineering design and manufacturing.

Financial Decision Making

The financial services industry has been a vigorous user of expert system techniques. Advisory programs have been created to assist bankers in determining whether to make loans to businesses and individuals. Insurance companies have used expert systems to assess the risk presented by the customer and to determine a price for the insurance. A typical application in the financial markets is in foreign exchange trading.

Knowledge Publishing

This is a relatively new, but also potentially explosive area. The primary function of the expert system is to deliver knowledge that is relevant to the user's problem, in the context of the user's problem. The two most widely distributed expert systems in the world are in this category. The first is an advisor which counsels a user on appropriate grammatical usage in a text. The second is a tax advisor that accompanies a tax preparation program and advises the user on tax strategy, tactics, and individual tax policy.

Process Monitoring and Control

Systems falling in this class analyze real-time data from physical devices with the goal of noticing anomalies, predicting trends, and controlling for both optimality and failure correction. Examples of real-time systems that actively monitor processes can be found in the steel making and oil refining industries.

Design and Manufacturing

These systems assist in the design of physical devices and processes, ranging from high-level conceptual design of abstract entities all the way to factory floor configuration of manufacturing processes.

Unit 14

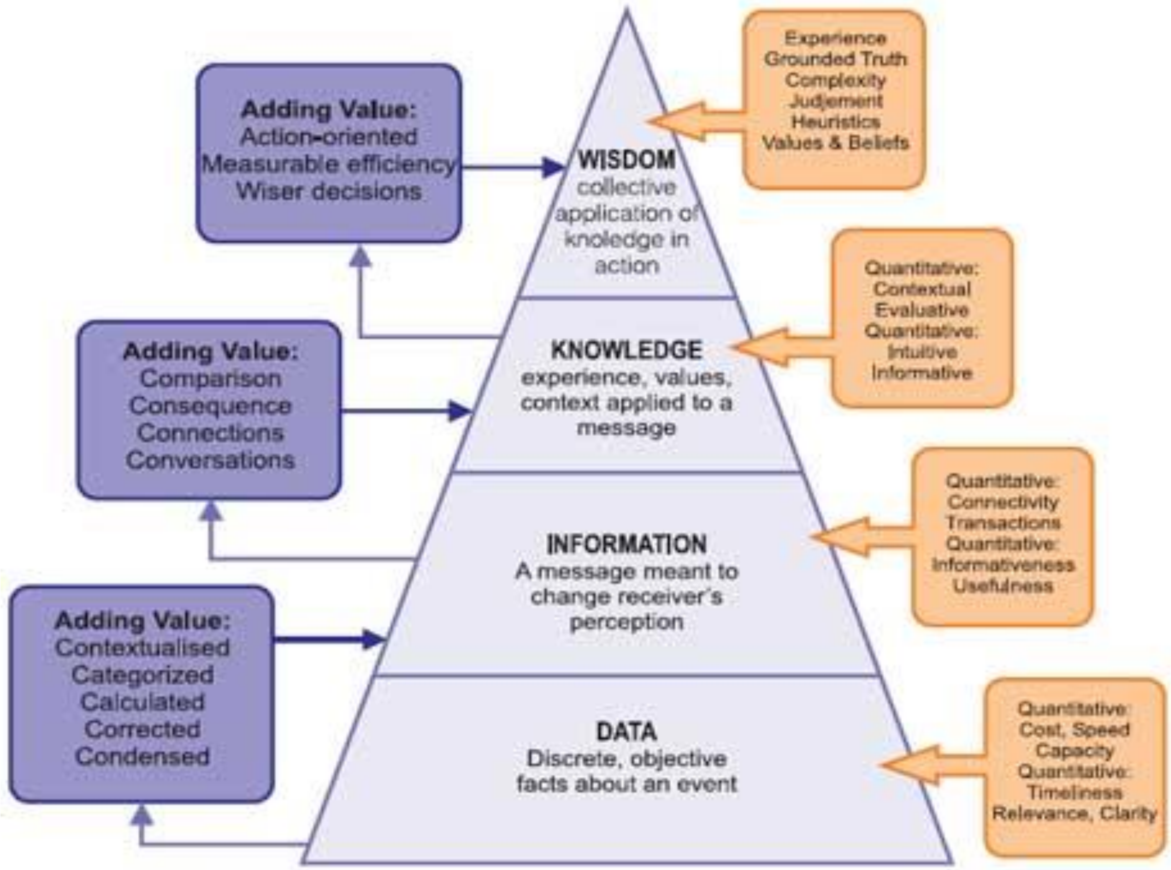
14. Data, Information and Knowledge

The concept of knowledge has been discussed for centuries and in the works of the ancient Greek philosophers, knowledge originates with people. Plato, for instance, put forward the idea that correct belief can be turned into knowledge by fixing it through the means of reason or a cause. Aristotle thought that knowledge of a thing involved understanding it in terms of the reasons for it. In Western philosophy knowledge is seen as abstract, universal, impartial and rational. It is considered as a stand-alone artifact (a physical record) that could be captured in technology and which will be truthful in its essence [4]. This understanding of knowledge affected, to a great extent, the nature of the first KM tools developed during the 90s. Most tools and KM models during this period tried to manage knowledge as an artifact rather than as an element deeply rooted in human understanding, human behaviour and social interactions at work. According to research, the majority of the first generation of KM tools failed, or at least did not fulfill their initial aims, due to the lack of focus on human factors. Knowledge has a far more complex nature than simple data and information and requires the active contribution of people to manage knowledge systems. Therefore, for proper KM implementation it is essential to clarify at an early stage, the main differences between data, information and knowledge.

The academic community has spent years discussing and clarifying what constitutes data, information and knowledge. Variations emerge in the definitions and the basic terminology used depending on the background of the author and the specific aims he pursues.

The relationship between data, information, knowledge and wisdom form a pyramid. The pyramid has data as its base, followed in the hierarchy by information, then knowledge, with wisdom at the top.

Figure 1.3 (1) below shows the relationships between data, information knowledge and wisdom.



- **Data** represents unorganized and unprocessed facts.
  - Usually data is static in nature.
  - It can represent a set of discrete facts about events.
  - Data is a prerequisite to information.
  - An organization sometimes has to decide on the nature and volume of data that is required for creating the necessary information.
- **Information**
  - Information can be considered as an aggregation of data (processed data) which makes decision making easier.
  - Information has usually got some meaning and purpose.
- **Knowledge**
  - By knowledge we mean *human understanding of a subject matter that has been acquired through proper study and experience*.
  - Knowledge is usually based on learning, thinking, and proper understanding of the problem area.
  - Knowledge is not information and information is not data.
  - Knowledge is derived from information in the same way information is derived from data.
  - We can view it as an understanding of information based on its perceived importance or relevance to a problem area.

- It can be considered as the integration of human perceptive processes that helps them to draw meaningful conclusions.

Unit 15

15. Knowledge Representation

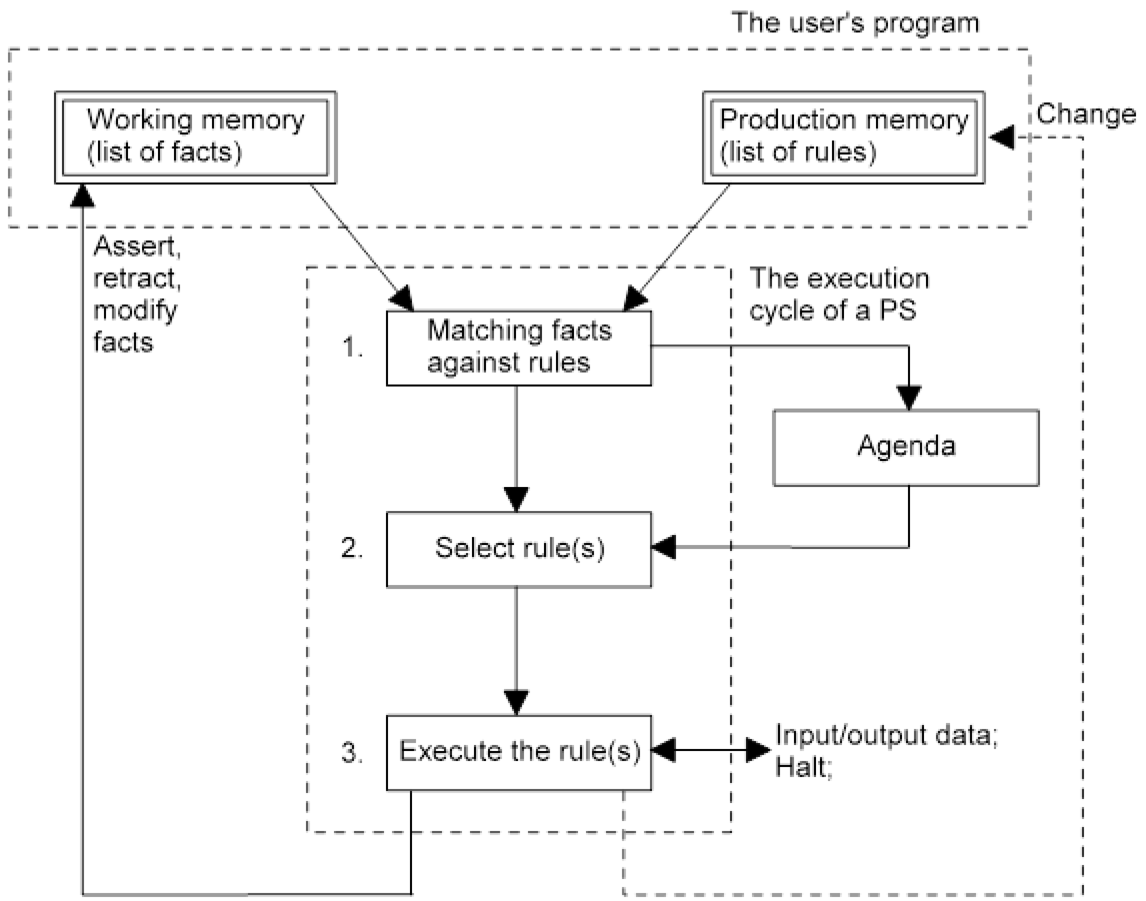
15.1 Production rules

- A production system is a model of cognitive processing, consisting of a collection of rules(called production rules, or just productions). Each rule has two parts: a condition part and an action part.
- The meaning of the rule is that when the condition holds true, then the action is taken.
  - Rule1: if temperature < 20°C → turn-on heating.
  - Rule2: if temperature > 20°C → turn-off heating.

  When the room temperature is below 20°C, the condition part of Rule1 is true, so the thermostat takes the action specified by the rule and turns on the heating. (The rule is said to fire
- A Production Rule System emulates human reasoning using a set of 'productions'
- Productions have two parts
  - Sensory precondition ("IF" part)
  - Action ("THEN" part)
- When the state of the 'world' matches the IF part, the production is *fired*, meaning the action is executed
  - The 'world' is the set of data values in the system's *working memory*
  - For a clinical expert systems, this is usually data about a patient, which, ideally, has come from (and may go back to) an electronic medical record, or it may be entered interactively (or usually a little of each)
- So production rules link *facts* ("IF" parts, also called *antecedents*) to *conclusions* ("THEN" parts, also called *consequents*)

  A production system consists of:
  - Working memory (facts memory)
  - Production rules memory
  - Inference engine, it cycles through three steps:
    - match facts against rules
    - select a rule

- **execute the rule**



15.2 Semantic networks

Semantic networks are an alternative to predicate logic as a form of knowledge representation. The idea is that we can store our knowledge in the form of a graph, with nodes representing objects in the world, and arcs representing relationships between those objects.

The major idea is that:

· The meaning of a concept comes from its relationship to other concepts, and that,

· The information is stored by interconnecting nodes with labeled arcs.

For example, the following:

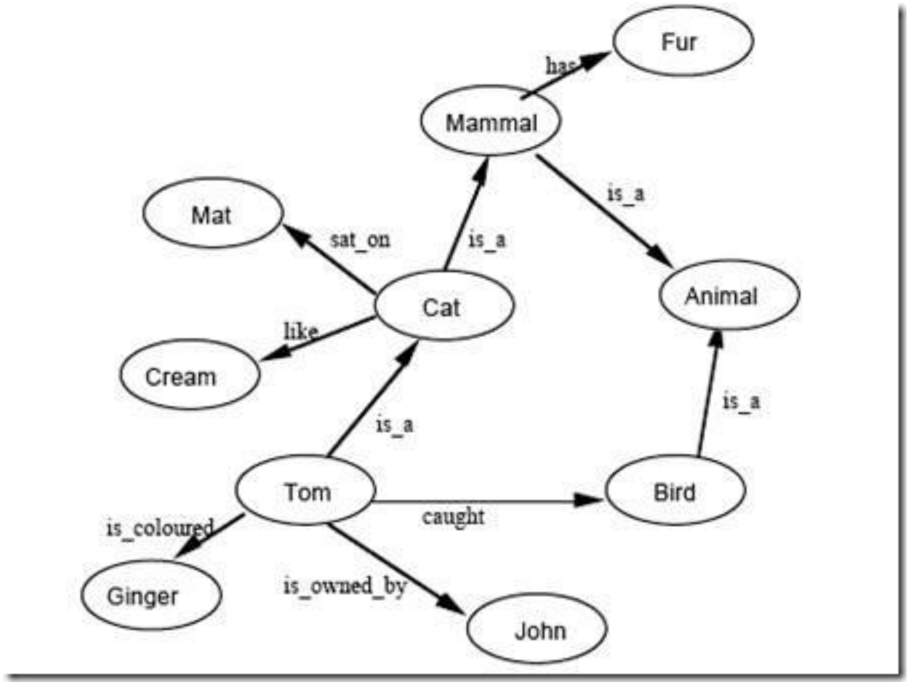


Figure 6.1: An example of Semantic Networks

is intended to represent the data:

Tom is a cat.

Tom caught a bird.

Tom is owned by John.

Tom is ginger in colour.

Cats like cream.

The cat sat on the mat.

A cat is a mammal.

A bird is an animal.

All mammals are animals.

Mammals have fur.

It is argued that this form of representation is closer to the way humans structure knowledge by building mental links between things than the predicate logic we considered earlier. Note in particular how all the information about a particular object is concentrated on the node representing that object, rather than scattered around several clauses in logic. There is, however, some confusion here which stems from the imprecise nature of semantic nets. A particular problem is that we haven't distinguished between nodes representing classes of things, and nodes representing individual objects. So, for example, the node labeled Cat represents both the single (nameless) cat who sat on the mat, and the whole class of cats to which Tom belongs, which are mammals and which like cream. The is_a link has two different meanings – it can mean that one object is an individual item from a class, for example Tom is a member of the class of cats, or that one class is a subset of another, for example, the class of cats is a subset of the class of mammals. This confusion does not occur in logic, where the use of quantifiers, names and predicates makes it clear what we mean so:

Tom is a cat is represented by Cat(Tom)

The cat sat on the mat is represented by $\exists x \exists y(Cat(x) \wedge Mat(y) \wedge SatOn(x,y))$

A cat is a mammal is represented by $\forall x(Cat(X) \rightarrow Mammal(x))$

We can clean up the representation by distinguishing between nodes representing individual or instances, and nodes representing *classes*. The is_a link will only be used to show an individual belonging to a class. The link representing one class being a subset of another will be labeled a_kind_of, or ako for short. The names instance and subclass are often used in the place of is_a and ako, but we will use these terms with a slightly different meaning in the section on Frames below.

Note also the modification which causes the link labeled is_owned_by to be reversed in direction. This is in order to avoid links representing passive relationships. In general a passive sentence can be replaced by an active one, so "Tom is owned by John" becomes "John owns Tom". In general the rule which converts passive to active in English converts sentences of the form "X is Yed by Z" to "Z Ys X". This is just an example (though often used for illustration) of the much more general principle of looking beyond the immediate *surface structure* of a sentence to find its *deep structure*.

The revised semantic net is:

:

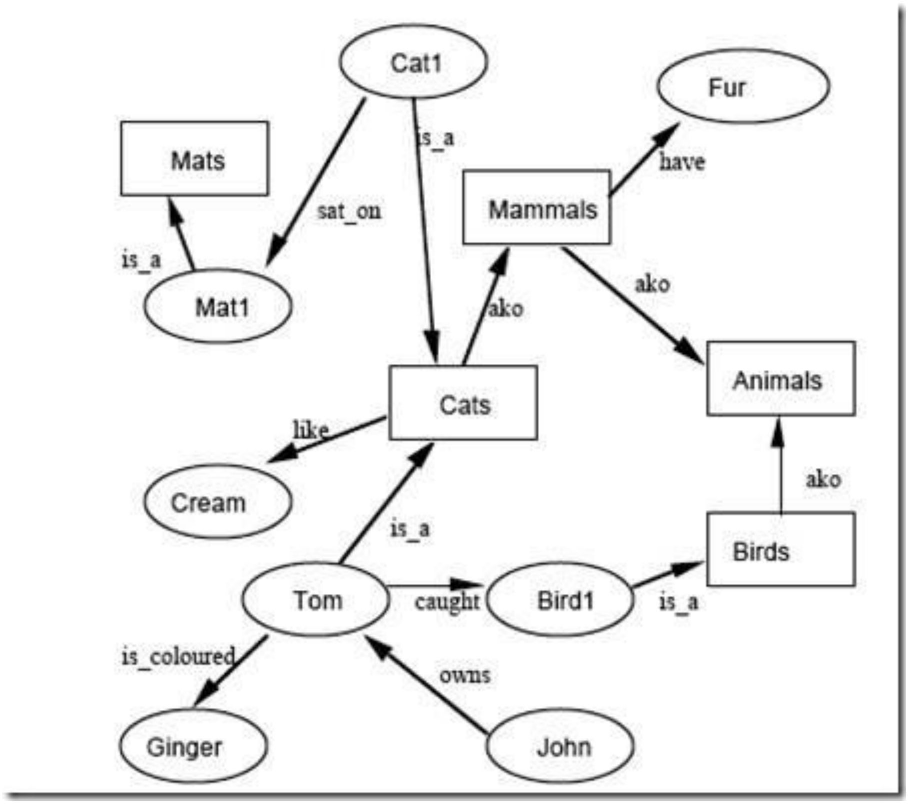


Figure 6.2: A Revised Semantic Networks

Note that where we had an unnamed member of some class, we have had to introduce a node with an invented name to represent a particular member of the class. This is a process similar to the Skolemisation we considered previously as a way of dealing with existential quantifiers. For example, "Tom caught a bird" would be represented in logic by x (bird(x) ∧ caught(Tom,x)), which would be Skolemised by replacing the x with a Skolem constant; the same thing was done above where bird1 was the name given to the individual bird that Tom caught. There are still plenty of issues to be resolved if we really want to represent what is meant by the English phrases, or to be really clear about what the semantic net means, but we are getting towards a notation that can be used practically (one example of a thing we have skated over is how to deal with mass nouns like "fur" or "cream" which refer to things that come in amounts rather than individual objects).

A direct Prolog representation can be used, with classes represented by predicates, thus:

cat(tom).

cat(cat1).

mat(mat1).

sat_on(cat1,mat1).

bird(bird1).

caught(tom,bird1).

like(X,cream) :– cat(X).

mammal(X) :– cat(X).

has(X,fur) :– mammal(X).

animal(X) :– mammal(X).

animal(X) :– bird(X).

owns(john,tom).

is_coloured(tom,ginger).

So, in general, an is_a link between a class c and an individual m is represented by the fact c(m). An a_kind_of link between a subclass c and a superclass s is represented by s(X) :- c(X). If a property p with further arguments a_1, … ,a_n is held by all members of a class c, it is represented by p(X,a1,…,an) :- c(X). If a property p with further arguments a_1, … ,a_n is specified as held by an individual m, rather than a class to which m belongs, it is represented by p(m,a_1,…,a_n).

The physical attributes of a person can be represented as in Fig. 6.3.

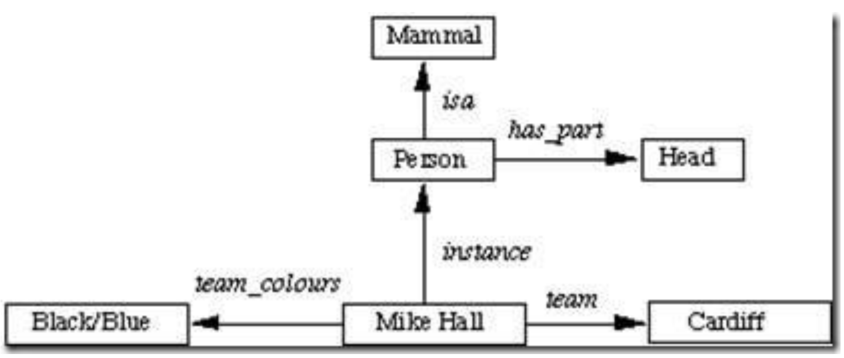


Figure 6.3: A Semantic Network for the attributes of a person

These values can also be represented in logic as: *isa(person, mammal), instance(Mike-Hall, person) team(Mike-Hall, Cardiff)*

As a more complex example consider the sentence: *John gave Mary the book*. Here we have several aspects of an event.

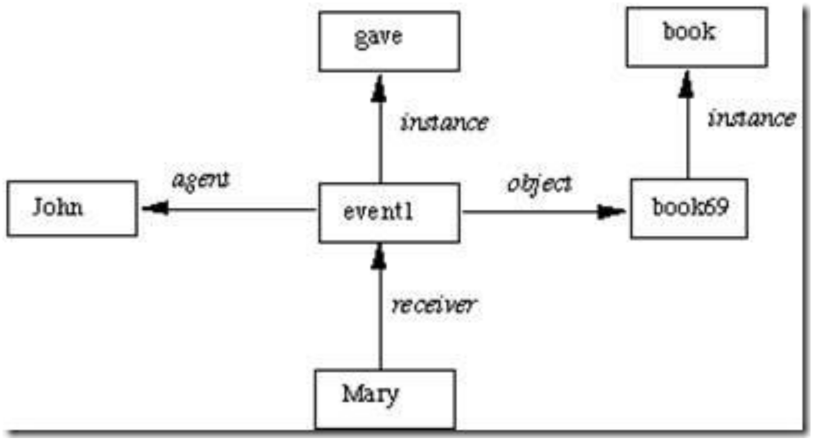


Figure 6.4: A Semantic Network for a Sentence

Inheritance

This Prolog equivalent captures an important property of semantic nets, that they may be used for a form of inference known as *inheritance*. The idea of this is that if an object belongs to a class (indicated by an is_a link) it *inherits* all the properties of that class. So, for example as we have a likes link between cats and cream, meaning "all cats like cream", we can infer that any object which has an is_a link to cats will like cream. So both Tom and Cat1 like cream. However, the is_coloured link is between Tom and ginger, not between cats and ginger, indicating that being ginger is a property of Tom as an individual, and not of all cats. We cannot say that Cat1 is ginger, for example; if we wanted to we would have to put another is_coloured link between Cat1 and ginger.

Inheritance also applies across the a_kind_of links. For example, any property of mammals or animals will automatically be a property of cats. So we can infer, for example, that Tom has fur, since Tom is a cat, a cat is a kind of mammal, and mammals have fur. If, for example, we had another subclass of mammals, say dogs, and we had, say, Fido is_a dog, Fido would inherit the property has fur from mammals, but not the property likes cream, which is specific to cats. This situation is shown in the diagram below:

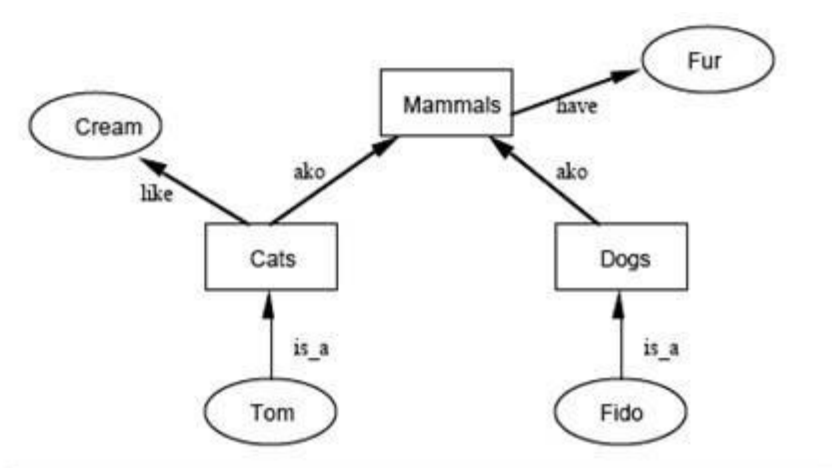


Figure 6.5: Semantic Networks showing Inheritance

Inheritance also provides a means of dealing with *default reasoning*, *e.g.* we could represent:

· Emu are birds.

· Typically birds fly and have wings.

· Emus run.

in the following Semantic net:

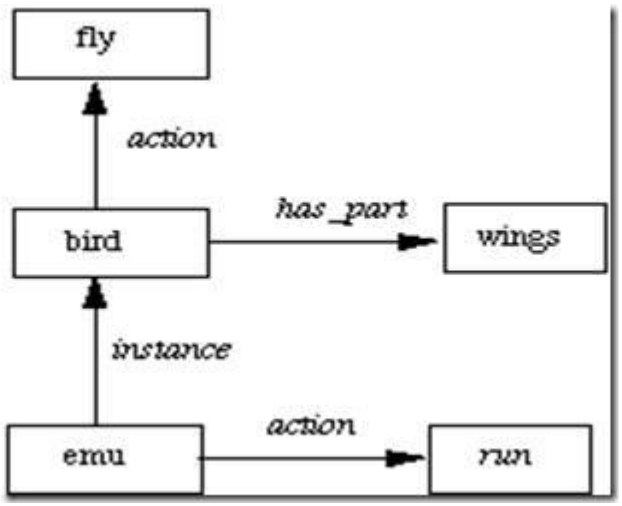


Figure 6.6: A Semantic Network for a Default Reasoning

15.3 Logic statements

Logic has a profound impact on computer-science. Some examples:

- Propositional logic – the foundation of computers and circuitry
- Databases – query languages
- Programming languages (e.g. prolog)
- Design Validation and verification
- AI (e.g. inference systems)
- Propositional Logic
- First Order Logic
- Higher Order Logic
- Temporal Logic

- A logic consists of *syntax* and *semantics*
- Syntax defines well formed sentences (Infix) Arithmetic
    - x+y=4
    - x4y+=
- Semantics defines "meaning" of sentences
- In logic, defines the *truth* of each sentence with respect to each *possible world*
- Possible World 1: x=1, y=3
- Possible World 2: x=2, y=1

- Sentences in Propositional Logic are defined in Backus-Naur Form (BNF):
    - A variable symbol (P,Q,R,…), and the constants True, False are correct sentences.
    - Connectives: ("Sentence-Forming Operators")
        - $\wedge$ conjunction , and
        - $\vee$ disjunction , or
        - $\neg$ negation "not", "it is not the case that"
        - $\rightarrow$ implication,
        - $\leftrightarrow$ equivalent to or bioconditional

The five logical connectives:

P	Q	$\neg P$	$P \wedge Q$	$P \vee Q$	$P \Rightarrow Q$	$P \Leftrightarrow Q$
False	*False*	*True*	*False*	*False*	*True*	*True*
False	*True*	*True*	*False*	*True*	*True*	*False*
True	*False*	*False*	*False*	*True*	*False*	*False*
True	*True*	*False*	*True*	*True*	*True*	*True*

A complex sentence:

P	H	$P \vee H$	$(P \vee H) \wedge \neg H$	$((P \vee H) \wedge \neg H) \Rightarrow P$
False	*False*	*False*	*False*	*True*
False	*True*	*True*	*False*	*True*
True	*False*	*True*	*True*	*True*
True	*True*	*True*	*False*	*True*

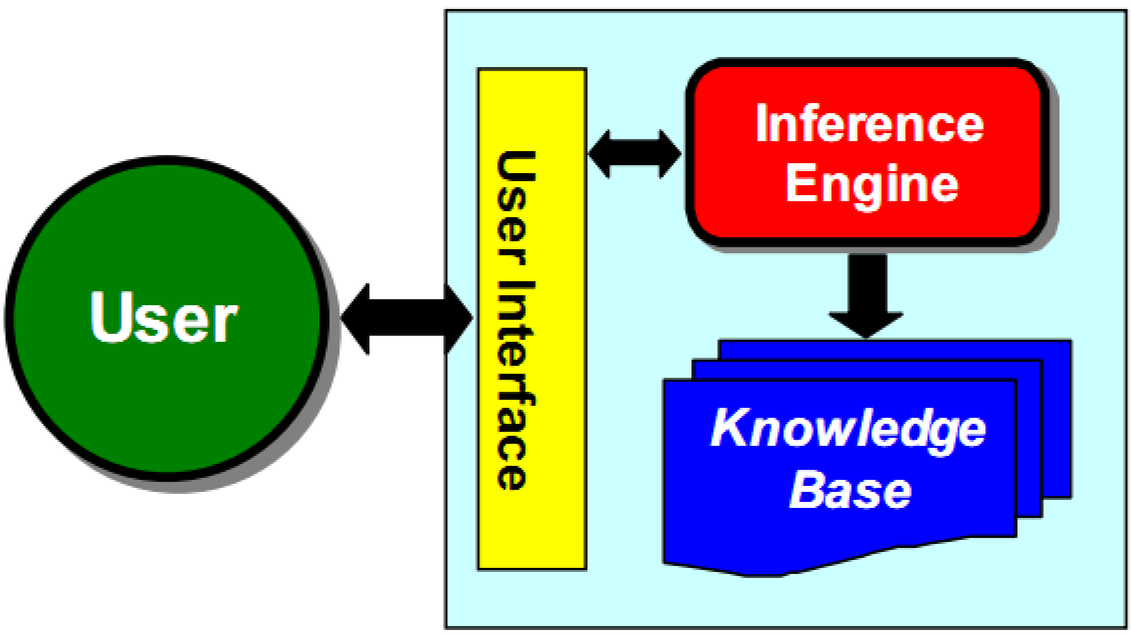
Unit 16

16. Inference, Explanations, and Uncertainty

An inference engine for rule based expert systems which forms part of the EXPRES system is developed and presented. It is shown to be universal, correct, and optimal with respect to time. In addition, a VLSI implementation of the system is proposed which allows automatic design of universal as well as special purpose expert systems on a chip.

Expert systems have become a popular method for representing large bodies of knowledge for a given field of expertise and solving problems by use of this knowledge. An expert system often consists of three parts, namely: a knowledge base, an inference engine, and a user interface

A dialogue is conducted by the user interface between the user and the system. The user provides information about the problem to be solved and the system then attempts to provide insights derived (or inferred) from the knowledge base. These insights are provided by the inference engine after examining the knowledge base. This interaction is illustrated by the picture in figure 1.



When rules are examined by the inference engine, actions are executed if the information supplied by the user satisfies the conditions in the rules. Two methods of inference often are used, forward and backward chaining. Forward chaining is a top-down method which takes facts as they become available and attempts to draw conclusions (from satisfied conditions in rules) which lead to actions being executed. Backward chaining is the reverse. It is a bottom-up procedure which starts with goals (or actions) and queries the user about information which may satisfy the conditions contained in the rules. It is a verification process rather than an exploration process. An example of backward chaining is MYCIN [vMS81], and an example of forward chaining is Expert [WK81]. A system which uses both is Prospector [DGH79].

```
if temperature > 100 and complexion = pale then
  cost := cost + 35, print("Patient has the flu."),
                     call Specialist, halt;
```

This has been a very brief introduction to the EXPRES system and the formulation of rule-based expert systems which can be designed using the system. There are more options which will are described elsewhere [Gr87], but this should be an adequate amount of material for the presentation of the inference engine in the next section.

- The inference engine is a computer program designed to produce a reasoning on rules.
- it is the "brain" that expert systems use to reason about the information in the knowledge base for the ultimate purpose of formulating new conclusions.

  Inference engines are considered to be a special case of reasoning engines, which can use more general methods of reasoning
- In order to produce a reasoning, it is based on logic.
- There are several kinds of logic: propositional logic, predicates of order 1 or more, epistemic logic, modal logic, temporal logic, fuzzy logic, etc.
- Propositional logic is the basic human logic, that is expressed in syllogisms.
- The expert system that uses propositional logic is also called a zeroth-order expert system.
- The inference engine can be described as a form of finite state machine with a cycle consisting of three action states: ***match rules, select rules,*** and ***execute rules*.**

- The Forward chaining and Backward chaining are two techniques often used by Inference engine for drawing inferences from the knowledge base.
- **Forward Chaining**

    - It is reasoning from facts to the conclusion which lead to actions being executed.
    - Forward chaining is a top-down method.
- **Backward chaining**
    - It starts with goals (or actions) and queries the user about information which may satisfy the conditions contained in the rules.

It is a bottom-up procedure.

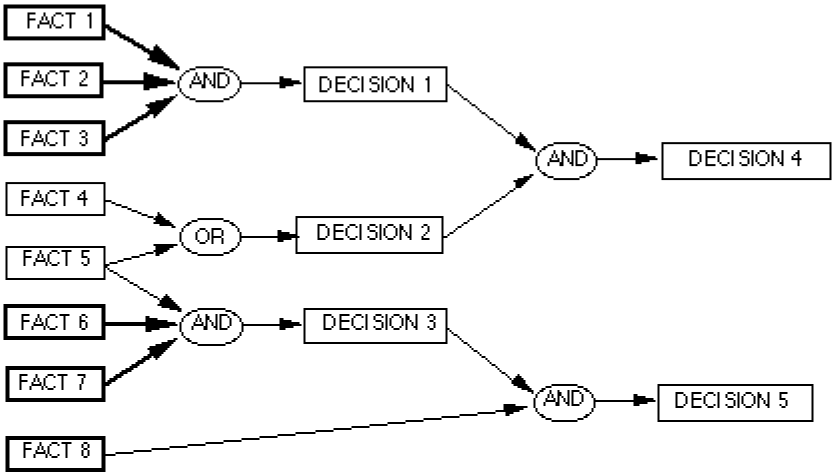
Forward Chaining And Backward Chaining Inference Techniques

Inference engines all match production rules (from the rule base) with facts (from the database). Two basic approaches are used to choose the order in which they are matched: forward chaining and backward chaining. Inference engines can use either or both. When both are used, backward chaining can be used initially, with forward chaining used as facts are added to the database.

Forward chaining

An inference technique that starts with the known data and works forward, matching the facts from the database with production rules from the rule base until no further rules can be fired.
Forward chaining is also referred to as data-driven reasoning.

	Rules	**Database**
Initial		A B C D E
Pass 1	3) A → X 4) C → L	
		A B C D E X L
Pass 2	2) X & B & E → Y	
		A B C D E X L Y
Pass 3	1) Y & D → Z	
		A B C D E X L Y Z
Pass 4		
		A B C D E X L Y Z

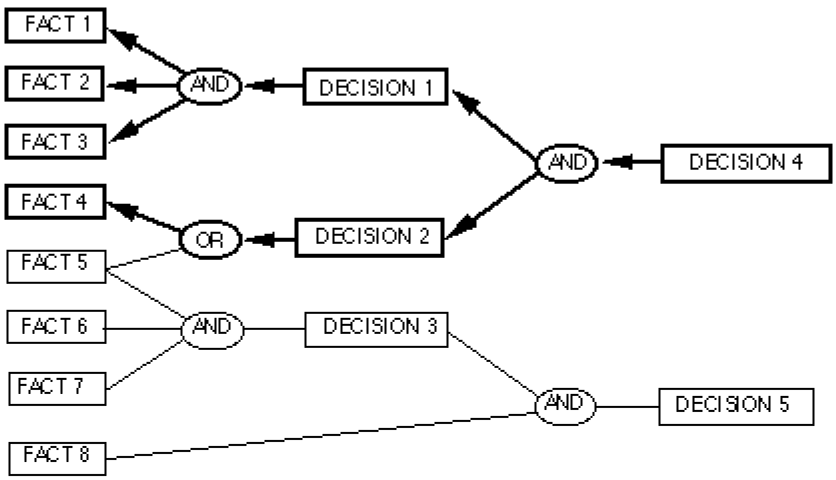


Backward chaining

An inference technique that starts with a hypothetical solution (the goal) and works backward, matching production rules from the rule base with facts from the database until the goal is either verified or proven wrong.

- Backward chaining is also referred to as goal-driven reasoning
- Backward chaining is also referred to as goal-driven reasoning.

	Rules	Database
Initial		A B C D E
Pass 1	1) Y & D → Z **?**	
		A B C D E
Pass 2	2) X & B & E → Y **?**	
		A B C D E
Pass 3	3) A → X	
		A B C D E X
Pass 2	2) X & B & E → Y **!**	
		A B C D E X Y
Pass 1	1) Y & D → Z **!**	
		A B C D E X Y Z



conflict resolution

A method for chosing which production rule to fire when more than one rule can be fired in a given cycle.

```
Rule 1    IF   the traffic light is green
          THEN the action is go

Rule 2    IF   the traffic light is red
          THEN the action is stop

Rule 3    IF   the traffic light is red
          THEN the action is go
```

Bayes Theorem (aka, Bayes Rule)

Bayes' theorem (also known as Bayes' rule) is a useful tool for calculating conditional probabilities. Bayes' theorem can be stated as follows:

Bayes' theorem. Let A_1, A_2, ... , A_n be a set of mutually exclusive events that together form the sample space S. Let B be any event from the same sample space, such that P(B) > 0. Then,

$$P(A_k \mid B) = \frac{P(A_k \cap B)}{P(A_1 \cap B) + P(A_2 \cap B) + . . . + P(A_n \cap B)}$$

Note: Invoking the fact that $P(A_k \cap B) = P(A_k)P(B \mid A_k)$, Baye's theorem can also be expressed as

$$P(A_k \mid B) = \frac{P(A_k) P(B \mid A_k)}{P(A_1) P(B \mid A_1) + P(A_2) P(B \mid A_2) + . . . + P(A_n) P(B \mid A_n)}$$

Unless you are a world-class statiscian, Bayes' theorem (as expressed above) can be intimidating. However, it really is easy to use. The remainder of this lesson covers material that can help you understand when and how to apply Bayes' theorem effectively.

Part of the challenge in applying Bayes' theorem involves recognizing the types of problems that warrant its use. You should consider Bayes' theorem when the following conditions exist.

- The sample space is partitioned into a set of mutually exclusive events { A_1, A_2, . . . , A_n }.
- Within the sample space, there exists an event B, for which P(B) > 0.
- The analytical goal is to compute a conditional probability of the form: $P(A_k \mid B)$.
- You know at least one of the two sets of probabilities described below.
  - $P(A_k \cap B)$ for each A_k
  - $P(A_k)$ and $P(B \mid A_k)$ for each A_k

Bayes' theorem describes the relationships that exist within an array of simple and conditional probabilities. For example: Suppose there is a certain disease randomly found in one-half of one percent (.005) of the general population. A certain clinical blood test is 99 percent (.99) effective in detecting the presence of this disease; that is, it will yield an accurate positive result in 99 percent of the cases where the disease is actually present. But it also yields false-positive results in 5 percent (.05) of the cases where the disease is not present. The following table shows (in red) the probabilities that are stipulated in the example and (in blue) the probabilities that can be inferred from the stipulated information:

$P_{(A)} = .005$	the probability that the disease will be present in any particular person
$P_{(\sim A)} = 1 - .005 = .995$	the probability that the disease will not be present in any particular person
$P_{(B\|A)} = .99$	the probability that the test will yield a positive result [B] **if** the disease is present [A]
$P_{(\sim B\|A)} = 1 - .99 = .01$	the probability that the test will yield a negative result [~B] **if** the disease is present [A]
$P_{(B\|\sim A)} = .05$	the probability that the test will yield a positive result [B] **if** the disease is not present [~A]
$P_{(\sim B\|\sim A)} = 1 - .05 = .95$	the probability that the test will yield a negative result [~B] **if** the disease is not present [~A]

Given this information, Bayes' theorem allows for the derivation of the two simple probabilities

$P_{(B)} = \left[P_{(B\|A)} \times P_{(A)}\right] + \left[P_{(B\|\sim A)} \times P_{(\sim A)}\right]$ $= \left[.99 \times .005\right] + \left[.05 \times .995\right] = .0547$	the probability of a positive test result [B], irrespective of whether the disease is present [A] or not present [~A]
$P_{(\sim B)} = \left[P_{(\sim B\|A)} \times P_{(A)}\right] + \left[P_{(\sim B\|\sim A)} \times P_{(\sim A)}\right]$ $= \left[.01 \times .005\right] + \left[.95 \times .995\right] = .9453$	the probability of a negative test result [~B], irrespective of whether the disease is present [A] or not present [~A]

which in turn allows for the calculation of the four remaining conditional probabilities

$P_{(A\|B)} = \left[P_{(B\|A)} \times P_{(A)}\right] / P_{(B)}$ $= \left[.99 \times .005\right] / .0547 = .0905$	the probability that the disease is present [A] **if** the test result is positive [B] (i.e., the probability that a positive test result will be a true positive)
$P_{(\sim A\|B)} = \left[P_{(B\|\sim A)} \times P_{(\sim A)}\right] / P_{(B)}$ $= \left[.05 \times .995\right] / .0547 = .9095$	the probability that the disease is not present [~A] **if** the test result is positive [B] (i.e., the probability that a positive test result will be a false positive)
$P_{(\sim A\|\sim B)} = \left[P_{(\sim B\|\sim A)} \times P_{(\sim A)}\right] / P_{(\sim B)}$	the probability that the disease is absent [~A] **if** the test result is negative [~B] (i.e., the probability that a negative test

= [.95 x .995] / .9453 = .99995	result will be a true negative)
$P_{(A\|\sim B)}$ = [$P_{(\sim B\|A)}$ x $P_{(A)}$] / $P_{(\sim B)}$ = [.01 x .005] / .9453 = .00005	the probability that the disease is present [A] **if** the test result is negative [~B] (i.e., the probability that a negative test result will be a false negative)

To perform calculations using Bayes' theorem, enter the probability for one or the other of the items in each of the following pairs (the remaining item in each pair will be calculated automatically). A probability value can be entered as either a decimal fraction such as .25 or a common fraction such as 1/4. Whenever possible, it is better to enter the common fraction rather than a rounded decimal fraction: 1/3 rather than .3333; 1/6 rather than .1667; and so forth.

$P_{(A)}$ **or** $P_{(\sim A)}$
$P_{(B|\sim A)}$ **or** $P_{(\sim B|\sim A)}$
$P_{(B|A)}$ **or** $P_{(\sim B|A)}$

Sample Problem

Bayes' theorem can be best understood through an example. This section presents an example that demonstrates how Bayes' theorem can be applied effectively to solve statistical problems.

Example 1

Marie is getting married tomorrow, at an outdoor ceremony in the desert. In recent years, it has rained only 5 days each year. Unfortunately, the weatherman has predicted rain for tomorrow. When it actually rains, the weatherman correctly forecasts rain 90% of the time. When it doesn't rain, he incorrectly forecasts rain 10% of the time. What is the probability that it will rain on the day of Marie's wedding?

Solution: The sample space is defined by two mutually-exclusive events - it rains or it does not rain. Additionally, a third event occurs when the weatherman predicts rain. Notation for these events appears below.

- Event A_1. It rains on Marie's wedding.
- Event A_2. It does not rain on Marie's wedding.
- Event B. The weatherman predicts rain.

In terms of probabilities, we know the following:

- $P(A_1) = 5/365 = 0.0136985$ [It rains 5 days out of the year.]
- $P(A_2) = 360/365 = 0.9863014$ [It does not rain 360 days out of the year.]
- $P(B \mid A_1) = 0.9$ [When it rains, the weatherman predicts rain 90% of the time.]
- $P(B \mid A_2) = 0.1$ [When it does not rain, the weatherman predicts rain 10% of the time.]

We want to know $P(A_1 \mid B)$, the probability it will rain on the day of Marie's wedding, given a forecast for rain by the weatherman. The answer can be determined from Bayes' theorem, as shown below.

$$P(A_1 \mid B) = \frac{P(A_1)\,P(B \mid A_1)}{P(A_1)\,P(B \mid A_1) + P(A_2)\,P(B \mid A_2)}$$

$$P(A_1 \mid B) = (0.014)(0.9) / [\,(0.014)(0.9) + (0.986)(0.1)\,]$$

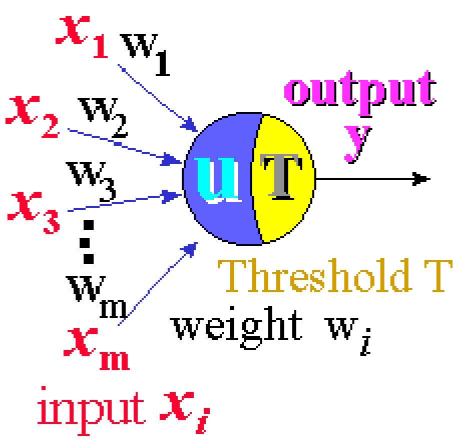
$$P(A_1 \mid B) = 0.111$$

Note the somewhat unintuitive result. Even when the weatherman predicts rain, it only rains only about 11% of the time. Despite the weatherman's gloomy prediction, there is a good chance that Marie will not get rained on at her wedding.

Unit- 18

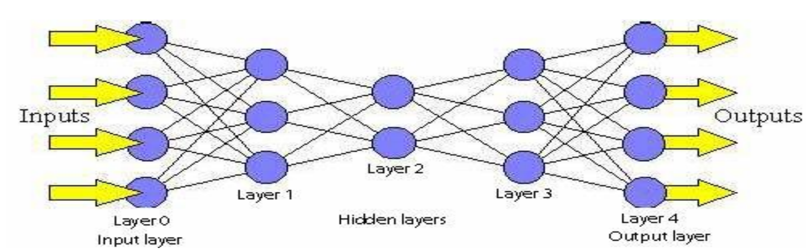
18. Introduction of Artificial Neural Network

Artificial neural network (ANN), usually called neural network (NN), is a system of mathematical model or computational model that is inspired by the structure and/or functional aspects of biological neural networks. There are many types of *artificial neural networks* (ANN). Each artificial neural network is a computational simulation of a biological neural network model. Artificial neural network models mimic the real life behavior of neurons and the electrical messages they produce between input (such as from the eyes or nerve endings in the hand), processing by the brain and the final output from the brain (such as reacting to light or from sensing touch or heat). There are other ANNs which are adaptive systems used to model things such as environments and population. Artificial neural network systems can be hardware and software based specifically built systems or purely software based and run in computer models.

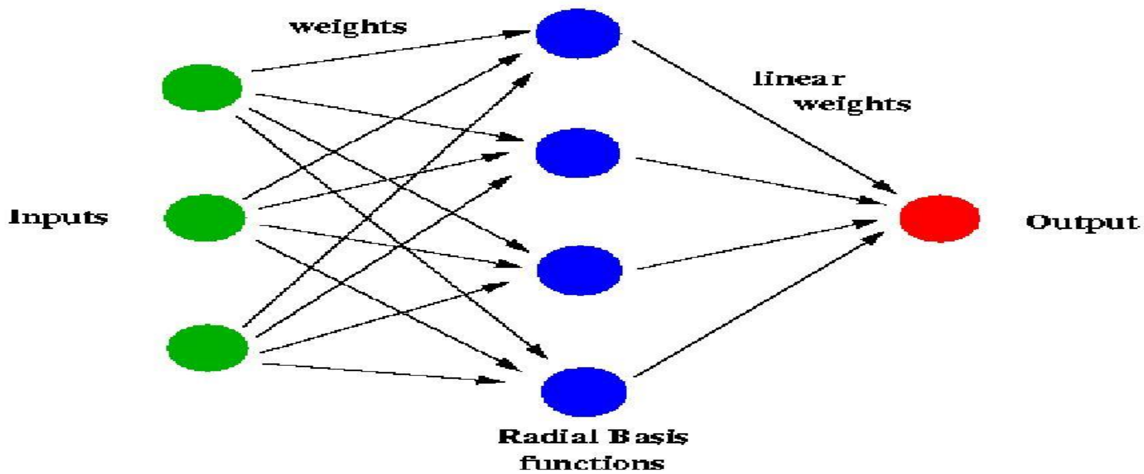


Types of Artificial Neural Networks (ANN)

- **Feed-forward Neural Network** – The feed-forward neural network was the first and arguably most simple type of artificial neural network devised. In this network the information moves in only one direction — forwards: From the input nodes data goes through the hidden nodes (if any) and to the output nodes. There are no cycles or loops in the network.

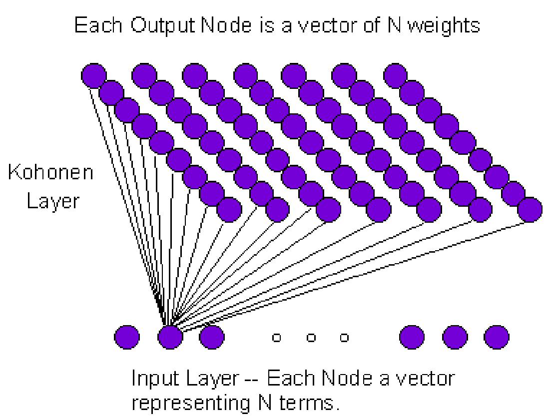


- **Radial Basis Function (RBF) Neural Network** – Radial basis functions are powerful techniques for interpolation in multidimensional space. A RBF is a function which has built into a distance criterion with respect to a center. RBF neural networks have the advantage of not suffering from local minima in the same way as Multi-Layer Perceptrons. RBF neural networks have the disadvantage of requiring goo

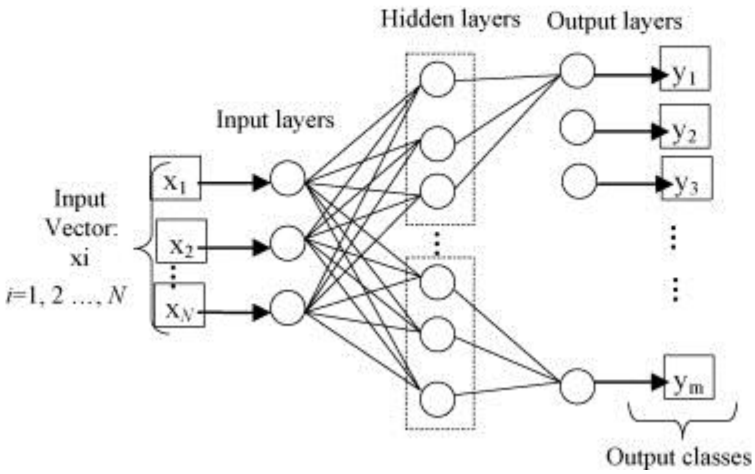


d coverage of the input space by radial basis functions.

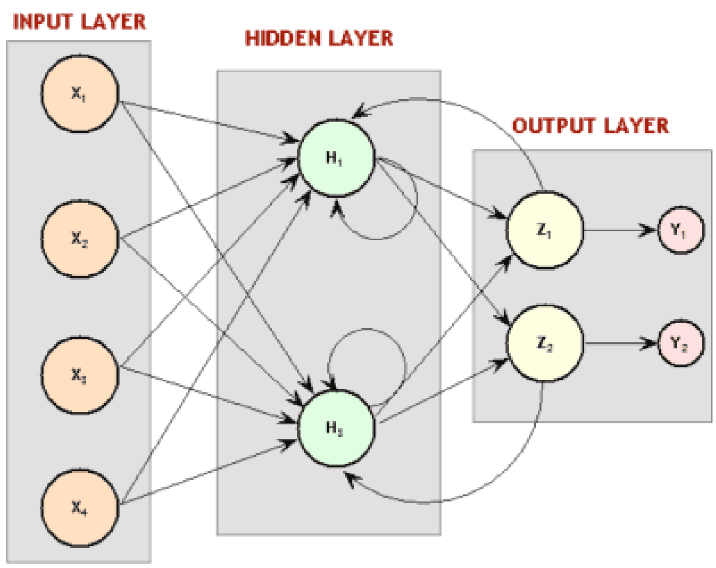
- **Kohonen Self-organizing Neural Network** – The self-organizing map (SOM) performs a form of unsupervised learning. A set of artificial neurons learn to map points in an input space to coordinates in an output space. The input space can have different dimensions and topology from the output space, and the SOM will attempt to preserve these.



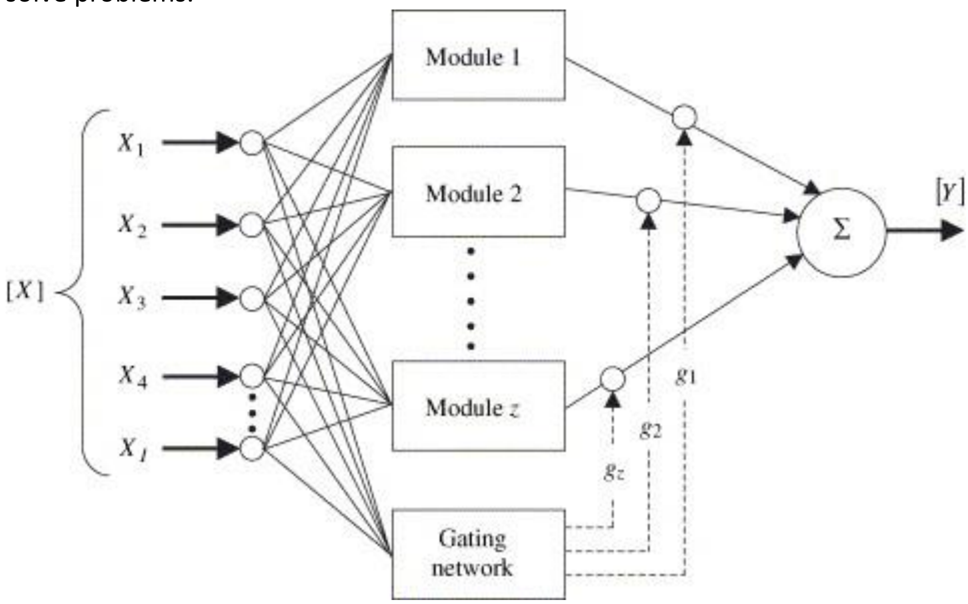
- **Learning Vector Quantization Neural Network** – Learning Vector Quantization (LVQ) can also be interpreted as a neural network architecture. In LVQ, prototypical representatives of the classes parameterize, together with an appropriate distance measure, a distance-based classification scheme.



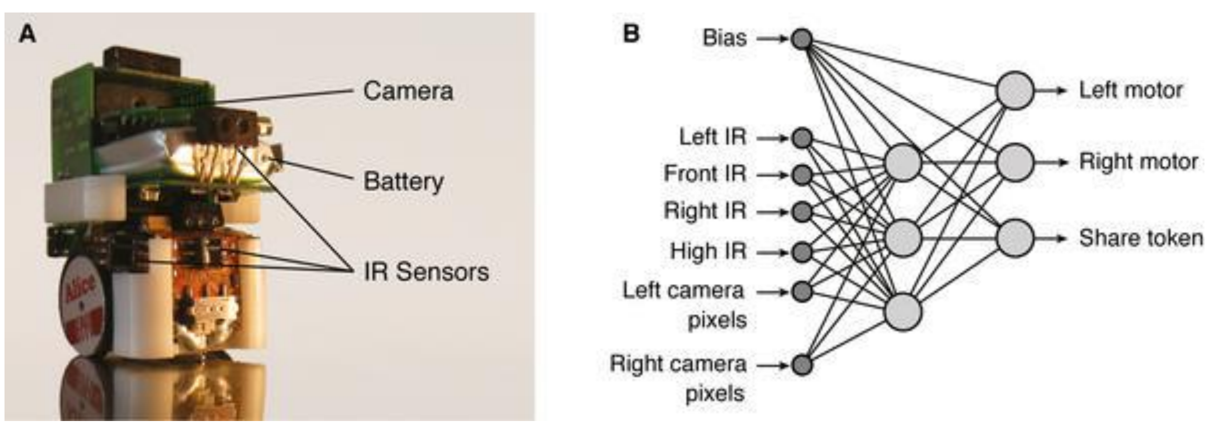
- **Recurrent Neural Networks** – Recurrent neural networks (RNNs) are models with bi-directional data flow. Recurrent neural networks can be used as general sequence processors. Various types of Recurrent neural networks are Fully recurrent network (**Hopfield network and Boltzmann machine**), Simple recurrent networks, Echo state network, Long short term memory network, Bi-directional RNN, Hierarchical RNN, and Stochastic neural networks.



- **Modular Neural Network** – Biological studies have shown that the human brain functions not as a single massive network, but as a collection of small networks. This realization gave birth to the concept of modular neural networks, in which several small networks cooperate or compete to solve problems.



- **Physical Neural Network** – A physical neural network includes electrically adjustable resistance material to simulate artificial synapses.



- **Other Special Types of Neural Networks**
  - **Holographic associative memory** – Holographic associative memory represents a family of analog, correlation-based, associative, stimulus-response memories, where information is mapped onto the phase orientation of complex numbers operating.
  - **Instantaneously Trained Neural Networks** – Instantaneously trained neural networks (ITNNs) were inspired by the phenomenon of short-term learning that seems to occur instantaneously.
  - **Spiking Neural Networks** – Spiking neural networks (SNNs) are models which explicitly take into account the timing of inputs. The network input and output are usually

represented as series of spikes (delta function or more complex shapes). SNNs have an advantage of being able to process information in the time domain (signals that vary over time).

- **Dynamic Neural Networks** – Dynamic neural networks not only deal with nonlinear multivariate behaviour, but also include (learning of) time-dependent behaviour such as various transient phenomena and delay effects.
- **Cascading Neural Networks** – Cascade Correlation is an architecture and supervised learning algorithm. Instead of just adjusting the weights in a network of fixed topology, Cascade-Correlation begins with a minimal network, then automatically trains and adds new hidden units one by one, creating a multi-layer structure.
- **Neuro-Fuzzy Neural Networks** – A neuro-fuzzy network is a fuzzy inference system in the body of an artificial neural network. Depending on the FIS type, there are several layers that simulate the processes involved in a fuzzy inference like fuzzification, inference, aggregation and defuzzification. Embedding an FIS in a general structure of an ANN has the benefit of using available ANN training methods to find the parameters of a fuzzy system.
- **Compositional Pattern-producing Neural Networks** – Compositional pattern-producing networks (CPPNs) are a variation of ANNs which differ in their set of activation functions and how they are applied. While typical ANNs often contain only sigmoid functions (and sometimes Gaussian functions), CPPNs can include both types of functions and many others.

Software

Several software on neural networks have been developed, to cite a few

Commercial Software : Statistica Neural Network, TNs2Server, DataEngine, Know Man Basic Suite, Partek, Saxon, ECANSE - Environment for Computer Aided Neural Software Engineering, Neuroshell, Neurogen, Matlab:Neural Network Toolbar.

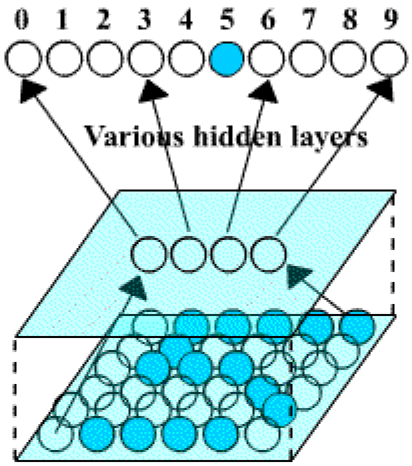
Freeware Software: Net II, Spider Nets Neural Network Library, NeuDC, Binary Hopfeild Net with free Java source, Neural shell, PlaNet, Valentino Computational Neuroscience Work bench, Neural Simulation language version-NSL, Brain neural network Simulator.

Unit 19

19.1 Neural Network Applications

Applications of Neural Network

Character Recognition - The idea of character recognition has become very important as handheld devices like the Palm Pilot are becoming increasingly popular. Neural networks can be used to recognize handwritten characters.



(Fig.1) A feed-forward network for character recognition.

The idea of using feedforward networks to recognize handwritten characters is rather straightforward. As in most supervised training, the bitmap pattern of the handwritten character is treated as an input, with the correct letter or digit as the desired output. Normally such programs require the user to train the network by providing the program with their handwritten patterns.

Image Compression - Neural networks can receive and process vast amounts of information at once, making them useful in image compression. With the Internet explosion and more sites using more images on their sites, using neural networks for image compression is worth a look.
Because neural networks can accept a vast array of input at once, and process it quickly, they are useful in image compression.

Bottleneck-type Neural Net Architecture for Image Compression

Here is a neural net architecture suitable for solving the image compression problem. This type of structure is referred to as a bottleneck type network, and consists of an input layer and an output layer of equal sizes, with an intermediate layer of smaller size in-between. The ratio of the size of the input layer to the

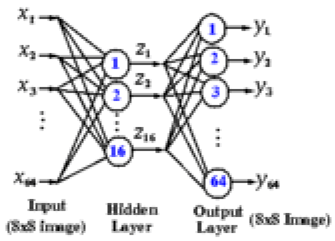


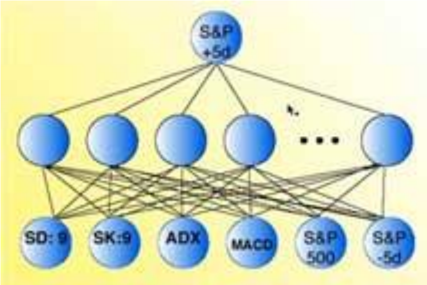
Image courtesy of:
http://neuron.eng.wayne.edu/bpImageCompression9PLUS/bp9PLUS.html

size of the intermediate layer is - of course - the compression ratio.

Stock Market Prediction - The day-to-day business of the stock market is extremely complicated. Many factors weigh in whether a given stock will go up or down on any given day. Since neural networks can examine a lot of information quickly and sort it all out, they can be used to predict stock prices. Neural networks have been touted as all-powerful tools in stock-market prediction. Companies such as MJ Futures claim amazing **199.2% returns** over a 2-year period using their neural network prediction methods. They also claim great ease of use; as technical editor John Sweeney said in a 1995 issue of "Technical Analysis of Stocks and Commodities," "you can skip developing complex rules (and redeveloping them as their effectiveness fades) . . . just define the price series and indicators you want to use, and the neural network does the rest."

These may be exaggerated claims, and, indeed, neural networks may be easy to use once the network is set up, but the setup and training of the network requires skill, experience, and patience. It's not all hype, though; neural networks have shown success at prediction of market trends.

The idea of stock market prediction is not new, of course. Business people often attempt to anticipate the market by interpreting external parameters, such as economic indicators, public opinion, and current political climate. The question is, though, if neural networks can discover trends in data that humans might not notice, and successfully use these trends in their predictions.



Good results have been achieved by Dean Barr and Walter Loick at LBS Capital Management using a relatively simple neural network with just 6 financial indicators as inputs. These inputs include the **ADX**, which indicates the average directional movement over the previous 18 days, the current value of the **S&P 500**, and the net change in the **S&P 500** value from five days prior (see David Skapura's book "Building Neural Networks," p129-154, for more detailed information).

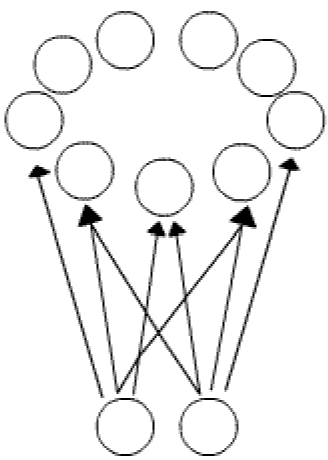
This is a simple back-propagation network of three layers, and it is trained and tested on a high volume of historical market data. The challenge here is not in the network architecture itself, but instead in the choice of variables and the information used for training. I could not find the accuracy rates for this network, but my source claimed it achieved "remarkable success" (this source was a textbook, not a NN-prediction-selling website!).

Even better results have been achieved with a back-propagated neural network with 2 hidden layers and many more than 6 variables. I have not been able to find more details on these network architectures, however; the companies that work with them seem to want to keep their details secret.

Traveling Saleman's Problem - Interestingly enough, neural networks can solve the traveling salesman problem, but only to a certain degree of approximation.

As in simple competitive networks, the weight vectors of the networks are randomly assigned at the beginning in SOM's. Another similarity is that SOM also identifies the winner (the perceptron whose weight vector is closest to the input vector) every time an input vector is presented. The only difference is that while in simple competitive networks only the winner learns, in SOM's all nodes learn, but the rate of learning varies inversely with the node's physical distance from the winner.

The Kohonen SOM used in solving the TSP has the following structure:



(Fig.3) A Kohonen net with a ring-shaped top layer.

Recall the elastic rubber-band model mentioned above, such a ring shaped map simulates a rubber-band if we consider the weight vectors as points on a plane. We can join these points together according to the position of their respective perceptron in the ring of the top layer of the network.

Suppose the coordinates of a city (x, y) is presented as the input vector of the network, the network will identify the weight vector closest to the city and move it and its neighbors closer to the city. In this manner, the weight vectors behave as points on a rubber band and each "learning" is analogous to pulling the closest point of the band towards a city. The rule that the amount of learning varies inversely with the physical distance between the node and the winner is what leads to the elastic property of the rubber band.

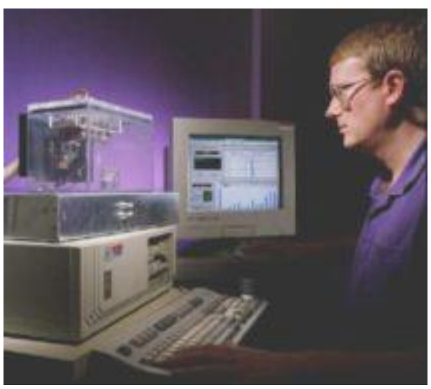
Medicine, Electronic Nose, Security, and Loan Applications - These are some applications that are in their proof-of-concept stage, with the acception of a neural network that will decide whether or not to grant a loan, something that has already been used more successfully than many humans.

Medicine

One of the areas that has gained attention is in cardiopulmonary diagnostics. The ways neural networks work in this area or other areas of medical diagnosis is by the comparison of many different models. A patient may have regular checkups in a particular area, increasing the possibility of detecting a disease or dysfunction.

The data may include heart rate, blood pressure, breathing rate, etc. to different models. The models may include variations for age, sex, and level of physical activity. Each individual's physiological data is compared to previous physiological data and/or data of the various generic models. The deviations from the norm are compared to the known causes of deviations for each medical condition. The neural network can learn by studying the different conditions and models, merging them to form a complete conceptual picture, and then diagnose a patient's condition based upon the models.

Electronic Noses



An actual electronic "nose"
Image courtesy Pacific Northwest Laboratory

The idea of a chemical nose may seem a bit absurd, but it has several real-world applications. The electronic nose is composed of a chemical sensing system (such as a spectrometer) and an artificial neural network, which recognizes certain patterns of chemicals. An odor is passed over the chemical sensor array, these chemicals are then translated into a format that the computer can understand, and the artificial neural network identifies the chemical.

A list at the Pacific Northwest Laboratory has several different applications in the environment, medical, and food industries.

Environment: identification of toxic wastes, analysis of fuel mixtures (7-11 example), detection of oil leaks, identification of household odors, monitoring air quality, monitoring factory emission, and testing ground water for odors.

Medical: The idea of using these in the medical field is to examine odors from the body to identify and diagnose problems. Odors in the breath, infected wounds, and body fluids all can indicate problems. Artificial neural networks have even been used to detect tuberculosis.

Food: The food industry is perhaps the biggest practical market for electronic noses, assisting or replacing entirely humans. Inspection of food, grading quality of food, fish inspection, fermentation control, checking mayonnaise for rancidity, automated flavor control, monitoring cheese ripening, verifying if orange juice is natural, beverage container inspection, and grading whiskey.

Security

One program that has already been started is the CATCH program. CATCH, an acronymn for Computer Aided Tracking and Characterization of Homicides. It learns about an existing crime, the location of the crime, and the particular characteristics of the offense. The program is subdivided into different tools, each of which place an emphasis on a certain characteristic or group of characteristics. This allows the user to remove certain characteristics which humans determine are unrelated.

Loans and credit cards

Loan granting is one area in which neural networks can aid humans, as it is an area not based on a predetermined and preweighted criteria, but answers are instead nebulous. Banks want to make as much money as they can, and one way to do this is to lower the failure rate by using neural networks to decide whether the bank should approve the loan. Neural networks are particularly useful in this area since no process will guarantee 100% accuracy. Even an 85-90% accuracy would be an improvement over the methods humans use.

In fact, in some banks, the failure rate of loans approved using neural networks is lower than that of some of their best traditional methods. Some credit card companies are now beginning to use neural networks in deciding whether to grant an application.

The process works by analyzing past failures and making current decisions based upon past experience. Nonetheless, this creates its own problems. For example, the bank or credit company must justify their decision to the applicant. The reason "my neural network computer recommended against it" simply isn't enough for people to accept. The process of explaining how the network learned and on what characteristics the neural network made its decision is difficult. As we alluded to earlier in the history of neural networks, self-modifying code is very difficult to debug and thus difficult to trace. Recording the steps it went through isn't enough, as it might be using conventional computing, because even the individual steps the neural network went through have to be analyzed by human beings, or possibly the network itself, to determine that a particular piece of data was crucial in the decision-making process.

Miscellaneous Applications - These are some very interesting (albeit at times a little absurd) applications of neural networks.

Neural networks have broad applicability to real world business problems. In fact, they have already been successfully applied in many industries.

Since neural networks are best at identifying patterns or trends in data, they are well suited for prediction or forecasting needs including:

- sales forecasting

- industrial process control
- customer research
- data validation
- risk management
- target marketing

But to give you some more specific examples; ANN are also used in the following specific paradigms: recognition of speakers in communications; diagnosis of hepatitis; recovery of telecommunications from faulty software; interpretation of multimeaning Chinese words; undersea mine detection; texture analysis; three-dimensional object recognition; hand-written word recognition; and facial recognition.

6.2 Neural networks in medicine

Artificial Neural Networks (ANN) are currently a 'hot' research area in medicine and it is believed that they will receive extensive application to biomedical systems in the next few years. At the moment, the research is mostly on modelling parts of the human body and recognising diseases from various scans (e.g. cardiograms, CAT scans, ultrasonic scans, etc.).

Neural networks are ideal in recognising diseases using scans since there is no need to provide a specific algorithm on how to identify the disease. Neural networks learn by example so the details of how to recognise the disease are not needed. What is needed is a set of examples that are representative of all the variations of the disease. The quantity of examples is not as important as the 'quantity'. The examples need to be selected very carefully if the system is to perform reliably and efficiently.

6.2.1 Modelling and Diagnosing the Cardiovascular System

Neural Networks are used experimentally to model the human cardiovascular system. Diagnosis can be achieved by building a model of the cardiovascular system of an individual and comparing it with the real time physiological measurements taken from the patient. If this routine is carried out regularly, potential harmful medical conditions can be detected at an early stage and thus make the process of combating the disease much easier.

A model of an individual's cardiovascular system must mimic the relationship among physiological variables (i.e., heart rate, systolic and diastolic blood pressures, and breathing rate) at different physical activity levels. If a model is adapted to an individual, then it becomes a model of the physical condition of that individual. The simulator will have to be able to adapt to the features of any individual without the supervision of an expert. This calls for a neural network.

Another reason that justifies the use of ANN technology, is the ability of ANNs to provide sensor fusion which is the combining of values from several different sensors. Sensor fusion enables the ANNs to learn complex relationships among the individual sensor values, which would otherwise be lost if the values were individually analysed. In medical modelling and diagnosis, this implies that even though each sensor in a set may be sensitive only to a specific physiological variable, ANNs are capable of detecting complex medical conditions by fusing the data from the individual biomedical sensors.

6.2.2 Electronic noses

ANNs are used experimentally to implement electronic noses. Electronic noses have several potential applications in telemedicine. Telemedicine is the practice of medicine over long distances via a communication link. The electronic nose would identify odours in the remote surgical environment. These identified odours would then be electronically transmitted to another site where an door generation system would recreate them. Because the sense of smell can be an important sense to the surgeon, telesmell would enhance telepresent surgery.
For more information on telemedicine and telepresent surgery click here.

6.2.3 Instant Physician

An application developed in the mid-1980s called the "instant physician" trained an autoassociative memory neural network to store a large number of medical records, each of which includes information on symptoms, diagnosis, and treatment for a particular case. After training, the net can be presented with input consisting of a set of symptoms; it will then find the full stored pattern that represents the "best" diagnosis and treatment.

6.3 Neural Networks in business

Business is a diverted field with several general areas of specialisation such as accounting or financial analysis. Almost any neural network application would fit into one business area or financial analysis.
There is some potential for using neural networks for business purposes, including resource allocation and scheduling. There is also a strong potential for using neural networks for database mining, that is, searching for patterns implicit within the explicitly stored information in databases. Most of the funded work in this area is classified as proprietary. Thus, it is not possible to report on the full extent of the work going on. Most work is applying neural networks, such as the Hopfield-Tank network for optimization and scheduling.

6.3.1 Marketing

There is a marketing application which has been integrated with a neural network system. The Airline Marketing Tactician (a trademark abbreviated as AMT) is a computer system made of various intelligent technologies including expert systems. A feedforward neural network is integrated with the AMT and was trained using back-propagation to assist the marketing control of airline seat allocations. The adaptive neural approach was amenable to rule expression. Additionaly, the application's environment changed rapidly and constantly, which required a continuously adaptive solution. The system is used to monitor and recommend booking advice for each departure. Such information has a direct impact on the profitability of an airline and can provide a technological advantage for users of the system. [Hutchison & Stephens, 1987]

While it is significant that neural networks have been applied to this problem, it is also important to see that this intelligent technology can be integrated with expert systems and other approaches to make a functional system. Neural networks were used to discover the influence of undefined interactions by the various variables. While these interactions were not defined, they were used by the neural system to develop useful conclusions. It is also noteworthy to see that neural networks can influence the bottom line.

6.3.2 Credit Evaluation

The HNC company, founded by Robert Hecht-Nielsen, has developed several neural network applications. One of them is the Credit Scoring system which increase the profitability of the existing model up to 27%. The HNC neural systems were also applied to mortgage screening. A neural network automated mortgage insurance underwritting system was developed by the Nestor Company. This system was trained with 5048 applications of which 2597 were certified. The data related to property and borrower qualifications. In a conservative mode the system agreed on the underwritters on 97% of the cases. In the liberal model the system agreed 84% of the cases. This is system run on an Apollo DN3000 and used 250K memory while processing a case file in approximately 1 sec.