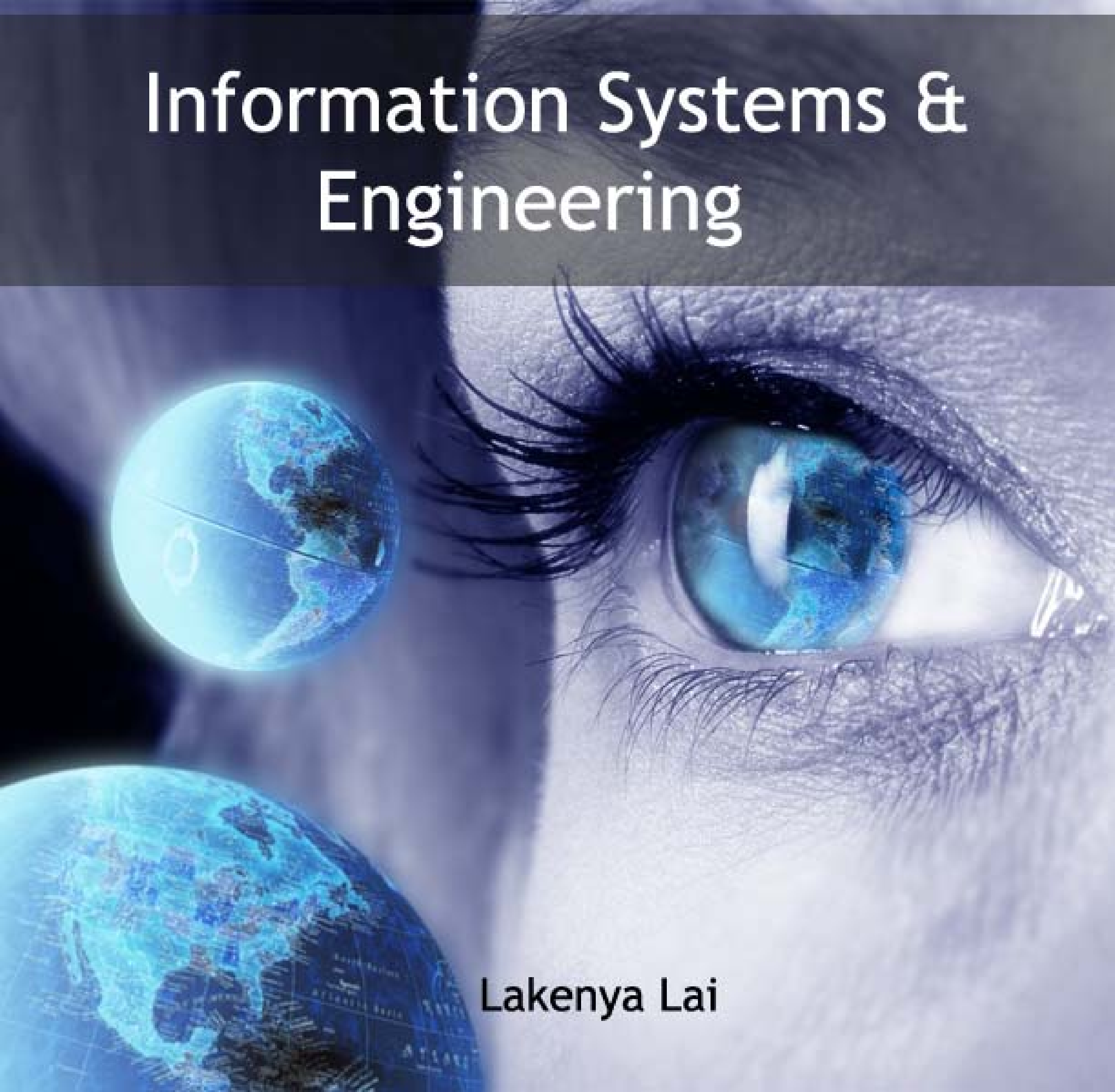
Information Systems & Engineering



Lakenya Lai

First Edition, 2012

ISBN 978-81-323-3074-5

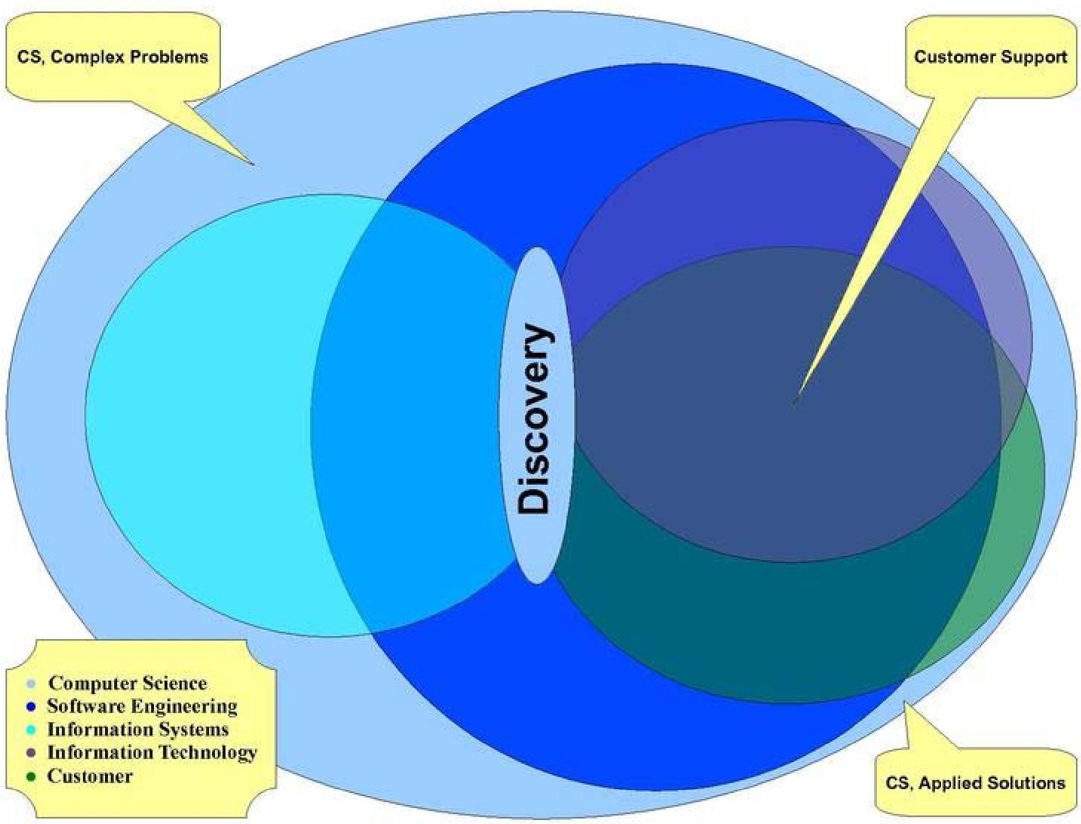
Published by:
Research World
4735/22 Prakashdeep Bldg,
Ansari Road, Darya Ganj,
Delhi - 110002
Email: info@wtbooks.com

Table of Contents

Chapter 1

Information Systems & Information Engineering

Information Systems



CS, SE, IS, IT, & Customer Venn Diagram where functionality spans left and design spans right stemming from discovery.

Information Systems (IS) is an academic/professional discipline bridging the business field and the well-defined computer science field that is evolving toward a new scientific area of study. An information systems discipline therefore is supported by the theoretical foundations of information and computations such that learned scholars have unique opportunities to explore the academics of various business models as well as related algorithmic processes within a computer science discipline. Typically, information systems or the more common *legacy* information systems include people, procedures, data, software, and hardware (by degree) that are used to gather and analyze digital information. Specifically computer-based information systems are complementary networks of hardware/software that people and organizations use to collect, filter, process, create, & distribute data (computing). *Computer* Information System(s) (CIS) is often a track within the computer science field studying computers and algorithmic processes, including their principles, their software & hardware designs, their applications, and their impact on society. Overall, an IS discipline emphasizes functionality over design.

As illustrated by the Venn Diagram on the right, the history of **information systems** coincides with the history of computer science that began long before the modern discipline of computer science emerged in the twentieth century. Regarding the circulation of information and ideas, numerous legacy information systems still exist today that are continuously updated to promote ethnographic approaches, to ensure data integrity, and to improve the social effectiveness & efficiency of the whole process. In general, information systems are focused upon processing information within organizations, especially within business enterprises, and sharing the benefits with modern society.

Overview

Silver et al. (1995) provided two views on (IS) and IS-centered view that includes software, hardware, data, people, and procedures. A second managerial view includes people, business processes and Information Systems.

There are various types of information systems, for example: transaction processing systems, office systems, decision support systems, knowledge management systems, database management systems, and office information systems. Critical to most information systems are information technologies, which are typically designed to enable humans to perform tasks for which the human brain is not well suited, such as: handling large amounts of information, performing complex calculations, and controlling many simultaneous processes.

Information technologies are a very important and malleable resource available to executives. Many companies have created a position of Chief Information Officer (CIO) that sits on the executive board with the Chief Executive Officer (CEO), Chief Financial Officer (CFO), Chief Operating Officer (COO) and Chief Technical Officer (CTO).The CTO may also serve as CIO, and vice versa. The Chief Information Security Officer (CISO), who focuses on information security management.

Definition

Silver et al. defined Information Systems as follows:

Information systems are implemented within an organization for the purpose of improving the effectiveness and efficiency of that organization. Capabilities of the information system and characteristics of the organization, its work systems, its people, and its development and implementation methodologies together determine the extent to which that purpose is achieved

The Discipline of Information Systems

Several IS scholars have debated the nature and foundations of Information Systems which has its roots in other reference disciplines such as Computer Science, Engineering, Mathematics, Management Science, Cybernetics, and others

The Impact on Economic Models

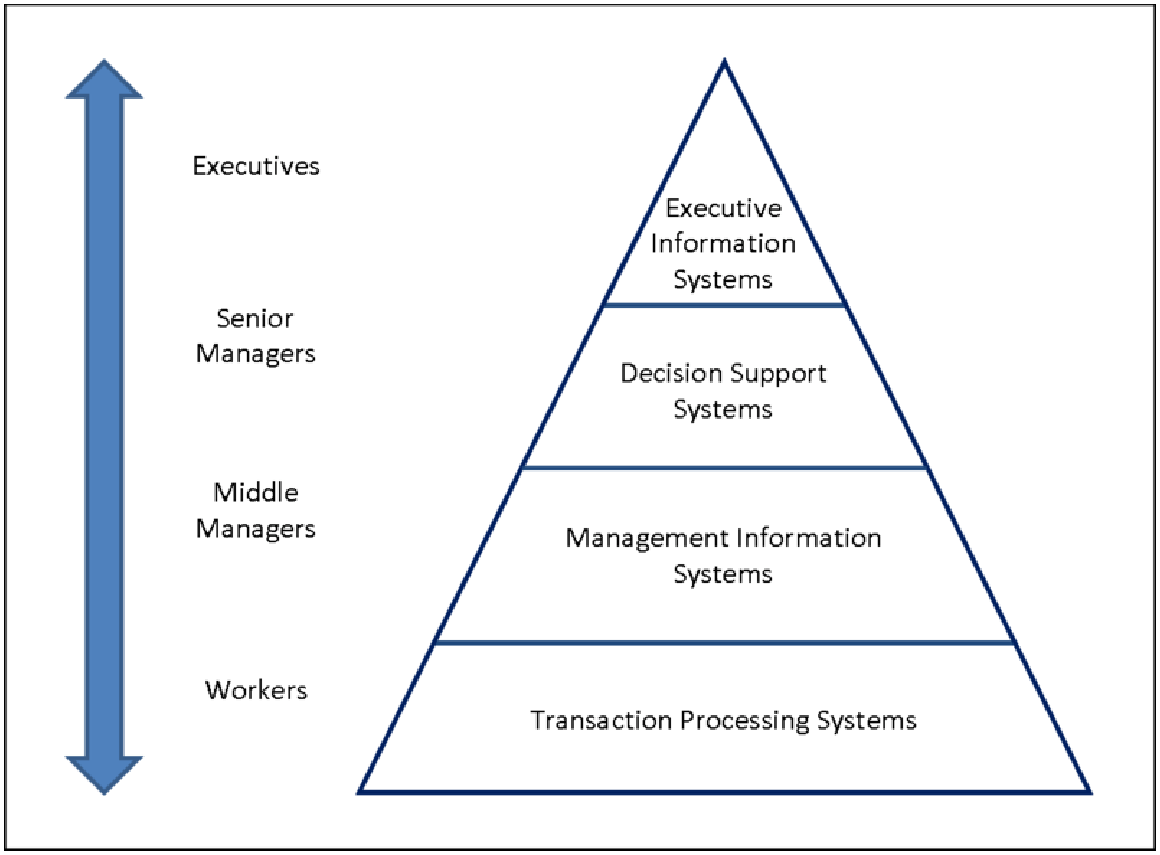
- Microeconomic theory model
- Transaction Cost theory
- Agency Theory

Differentiating IS from Related Disciplines

Similar to computer science, other disciplines can be seen as both related disciplines and foundation disciplines of IS. But, while there may be considerable overlap of the disciplines at the boundaries, the disciplines are still differentiated by the focus, purpose and orientation of their activities.

In a broad scope, the term **Information Systems** (IS) is a scientific field of study that addresses the range of strategic, managerial and operational activities involved in the gathering, processing, storing, distributing and use of information, and its associated technologies, in society and organizations. The term information systems is also used to describe an organizational function that applies IS knowledge in industry, government agencies and not-for-profit organizations. ***Information Systems*** often refers to the interaction between algorithmic processes and technology. This interaction can occur within or across organizational boundaries. An information system is not only the technology an organization uses, but also the way in which the organizations interact with the technology and the way in which the technology works with the organization’s business processes. Information systems are distinct from information technology (IT) in that an information system has an information technology component that interacts with the processes components.

Types of information systems



A four level pyramid model of different types of Information Systems based on the different levels of hierarchy in an organization

The 'classic' view of Information systems found in the textbooks of the 1980s was of a pyramid of systems that reflected the hierarchy of the organization, usually Transaction processing systems at the bottom of the pyramid, followed by Management information systems, Decision support systems and ending with Executive information systems at the top. Although the pyramid model remains useful, since it was first formulated a number of new technologies have been developed and new categories of information systems have emerged, some of which no longer fit easily into the original pyramid model.

Some examples of such systems are:

- Data warehouses
- Enterprise resource planning
- Enterprise systems
- Expert systems
- Geographic information system
- Global information system
- Office Automation

Information systems career pathways

Information Systems have a number of different areas of work:

- Information systems strategy
- Information systems management
- Information systems development
- Information systems security
- Information systems iteration
- Information system organization

There are a wide variety of career paths in the information systems discipline. "Workers with specialized technical knowledge and strong communications skills will have the best prospects. Workers with management skills and an understanding of business practices and principles will have excellent opportunities, as companies are increasingly looking to technology to drive their revenue."

Information systems development

Information technology departments in larger organizations tend to strongly influence information technology development, use, and application in the organizations, which may be a business or corporation. A series of methodologies and processes can be used in order to develop and use an information system. Many developers have turned and used a more engineering approach such as the System Development Life Cycle (SDLC) which is a systematic procedure of developing an information system through stages that occur in sequence. An Information system can be developed in house (within the organization) or outsourced. This can be accomplished by outsourcing certain components or the entire system. A specific case is the geographical distribution of the development team (Offshoring, Global Information System).

A computer based information system, following a definition of Langefors, is:

- a technologically implemented medium for recording, storing, and disseminating linguistic expressions,
- as well as for drawing conclusions from such expressions.

which can be formulated as a generalized information systems design mathematical program

Geographic Information Systems, Land Information systems and Disaster Information Systems are also some of the emerging information systems but they can be broadly considered as Spatial Information Systems. System development is done in stages which include:

- Problem recognition and specification
- Information gathering

- Requirements specification for the new system
- System design
- System construction
- System implementation
- Review and maintenance

Information systems research

Information systems research is generally interdisciplinary concerned with the study of the effects of information systems on the behavior of individuals, groups, and organizations. Hevner et al. (2004) categorized research in IS into two scientific paradigms including *behavioral science* which is to develop and verify theories that explain or predict human or organizational behavior and *design science* which extends the boundaries of human and organizational capabilities by creating new and innovative artifacts.

Salvatore March and Gerald Smith proposed a framework for researching different aspects of Information Technology including outputs of the research (research outputs) and activities to carry out this research (research activities). They identified research outputs as follows:

1. *Constructs* which are concepts that form the vocabulary of a domain. They constitute a conceptualization used to describe problems within the domain and to specify their solutions.
2. A *model* which is a set of propositions or statements expressing relationships among constructs.
3. A *method* which is a set of steps (an algorithm or guideline) used to perform a task. Methods are based on a set of underlying constructs and a representation (model) of the solution space.
4. An *instantiation* is the realization of an artifact in its environment.

Also research activities including:

1. *Build* an artifact to perform a specific task.
2. *Evaluate* the artifact to determine if any progress has been achieved.
3. Given an artifact whose performance has been evaluated, it is important to determine why and how the artifact worked or did not work within its environment. Therefore *theorize* and *justify* theories about IT artifacts.

Although Information Systems as a discipline has been evolving for over 30 years now, the core focus or identity of IS research is still subject to debate among scholars such as. There are two main views around this debate: a narrow view focusing on the IT artifact as the core subject matter of IS research, and a broad view that focuses on the interplay between social and technical aspects of IT that is embedded into a dynamic evolving context. A third view provided by calling IS scholars to take a balanced attention for both the IT artifact and its context.

Since information systems is an applied field, industry practitioners expect information systems research to generate findings that are immediately applicable in practice. However, that is not always the case. Often information systems researchers explore behavioral issues in much more depth than practitioners would expect them to do. This may render information systems research results difficult to understand, and has led to criticism.

To study an information system itself, rather than its effects, information systems models are used, such as EATPUT.

The international body of Information Systems researchers, the Association for Information Systems (AIS), and its Senior Scholars Forum Subcommittee on Journals (23 April 2007), proposed a 'basket' of journals that the AIS deems as 'excellent', and nominated: Management Information Systems Quarterly (MISQ), Information Systems Research (ISR), Journal of Association of Information Systems (JAIS), Journal of Management Information Systems (JMIS), European Journal of Information Systems (EJIS), and Information Systems Journal (ISJ).

Information Engineering

Information engineering (IE) or *information engineering methodology* (IEM) in software engineering is an approach to designing and developing information systems.

Overview

Information engineering methodology is an architectural approach to planning, analyzing, designing, and implementing applications within an enterprise. It aims to enable an enterprise to improve the management of its resources, including capital, people and information systems, to support the achievement of its business vision. It is defined as: *"An integrated and evolutionary set of tasks and techniques that enhance business communication throughout an enterprise enabling it to develop people, procedures and systems to achieve its vision"*.

Information engineering has many purposes, including organization planning, business re-engineering, application development, information systems planning and systems re-engineering.

History

Information engineering has a somewhat chequered history that follows two very distinct threads. It is said to have originated in Australia between 1976 and 1980, and appears first in the literature in 1981 in the Savant Institute publication 'Information engineering' by James Martin and Clive Finkelstein.

Information engineering first provided data analysis and database design techniques that could be used by database administrators (DBAs) and by systems analysts to develop database designs and systems based upon an understanding of the operational processing needs of organizations for the 1980s.

The Finkelstein thread evolved after 1980 into the data processing (DP)-driven variant of IE. From 1983 till 1986 IE evolved further into the business-driven variant of IE, which was intended to address a rapidly changing business environment. The then technical director, Charles M. Richter, from 1983 to 1987, played a significant role by revamping the IE methodology as well as designing the IE software product (user-data) which helped automate the IE methodology, opening the way to next generation Information Architecture.

The Martin thread was strategy-driven from the outset and from 1983 was focused on the possibility of automating the development process through the provision of techniques for business description that could be used to populate a data dictionary or encyclopedia that could in turn be used as source material for code generation. The Martin methodology provided a foundation for the CASE (computer-aided software engineering) tool industry. Martin himself had significant stakes in at least four CASE tool vendors - InTech (Excelerator), Higher Order Software, KnowledgeWare, originally Database Design Inc, (Information Engineering Workbench) and James Martin Associates, originally DMW and now Headstrong (the original designers of the Texas Instruments' Information Engineering Facility and the principal developers of the methodology).

At the end of the 1980s and early 1990s the Martin thread incorporated rapid application development (RAD) and business process reengineering (BPR) and soon after also entered the object oriented field.

Information engineering topics

IE variants

There are two variants of information engineering. These are called the DP-driven variant and the business-driven variant.

- DP-driven : The DP-driven variant of Information engineering was designed to enable IS Departments to develop information systems that satisfied the information needs of the 1980s - which was largely a DP-driven development environment. Most of the CASE tools available today support this DP-driven variant of IE.
- Business-driven: IE was extended into strategic business planning and developed the business-driven variant of information engineering. This variant was designed for rapid change in the client/server, object-oriented environment of the business-driven 1990s.

Business-driven IE is documented in the later books by Clive Finkelstein.

- Information Strategy Planning : The fundamental objective of Information Strategy Planning (ISP) is to develop a plan for implementing business systems to support business needs.
- Outline Business Area Analysis : The OBAA answers a range of questions related to implementation of a business area. Select tasks to include in a particular project that provide support for business decisions and objectives. Specific information needs and priorities for the business area are needed.
- Detailed Business Area Analysis : The purpose of a DBAA project is to provide detailed models as a solid basis for system design. The methodology helps find the right answers to the right questions. Applying the methodology is never an end in itself.
- Business System Design : The purpose of a Business System Design project is to specify all aspects of a system that are relevant to its users, in preparation for the technical design, construction, and installation of one or more closely related databases and systems. The key tasks are therefore structured to produce unambiguous consistent specifications, with the volume of detail necessary to make planning and technical design decisions.
- Technical Design : A Technical Design project prepares an implementation area for construction and installation. The key tasks are structured to produce a system and database that meet the user's acceptance criteria and are technically sound.
- Construction : The objective of the Construction stage is to produce a system, as defined in the technical specification, on time and within budget. The system should be of an acceptable quality, and contain all necessary operating and user procedures. The task is complete when the acceptance criteria for the business system are met.
- Transition : Transition is defined as the period during which newly developed procedures gradually replace or are interfaced with existing procedures. The execution of a Transition project obviously demands a thorough understanding of both the system to be installed and the systems to be replaced.

IE techniques

Some techniques that are used during an IE project are:

- Entity analysis : identifies all the things that the enterprise may want to hold data about. The analysis classifies all of the things into different entity types, revealing how they relate to each other. Which is being described in the entity model.
- Function analysis and process dependency : takes a function (a major business activity) of the enterprise and breaks it down into elementary business processes. From this, two diagrams are prepared: the process decomposition diagram, which shows the breakdown of a business function, and the process dependency diagram, which shows the interdependencies of business processes.
- Process logic analysis : describes the sequences of actions carried out by a business process and shows which data are used by each action.
- Entity type lifecycle analysis : describes the significant business changes to entities and confirm that processes have been modelled to effect these changes

- Matrix cross-checking : creates cross-references between data objects and processes to verify that they are necessary and complete.
- Normalization : provides a formal means of confirming the correctness of the entity model.
- Cluster analysis : helps define the scope of design areas for proposed business systems.
- Data flow and data analysis : makes a comparison possible between the business area models and the systems currently supporting this area, these current systems are analyzed using data flow and data analysis techniques.

Software tools

There are several tools supporting Information engineering

- Information engineering Facility (IEF) from Texas Instruments Software. This was subsequently sold to Sterling Software and then to Computer Associates. It still exists, in an evolved form within the Advantage suite. As of 2006 referred to as ALL:Fusion Gen, capable of generating J2EE and JAVA web applications in addition to legacy client/server and mainframe platforms.
- Information engineering Workbench (IEW) : Later renamed to Application Development Workbench (ADW) from KnowledgeWare. Knowledgeware was also acquired by Sterling Software. The product no longer exists.
- The business-driven variant of IE is supported by Visible Advantage, an Integrated CASE (I-CASE) tool and by Visible Advisor, a hypermedia Methodology product.
- Metastorm's ProVision product provides support for many types of modeling techniques using a repository based tool.
- Visio provides diagramming support to some of the Martin techniques.

Others included Bachman's Data Analyst, Excelerator and others.

Chapter 2

Geographic Information System

A **geographic information system (GIS)**, **geographical information system**, or **geospatial information system** is the system that captures, stores, analyzes, manages, and presents data with reference to geographic location data. In the simplest terms, GIS is the merging of cartography, statistical analysis, and database technology. GIS may be used in archaeology, geography, cartography, remote sensing, land surveying, public utility management, natural resource management, precision agriculture, photogrammetry, urban planning, emergency management, landscape architecture, navigation, aerial video, and localized search engines.

As GIS can be thought of as a system, it digitally creates and "manipulates" spatial areas that may be jurisdictional, purpose or application oriented for which a specific GIS is developed. Hence, a GIS developed for an application, jurisdiction, enterprise, or purpose may not be necessarily interoperable or compatible with a GIS that has been developed for some other application, jurisdiction, enterprise, or purpose. What goes beyond a GIS is a spatial data infrastructure (SDI), a concept that has no such restrictive boundaries.

Therefore, in a general sense, the term describes any information system that integrates, stores, edits, analyzes, shares, and displays geographic information for informing decision making. GIS applications are tools that allow users to create interactive queries (user-created searches), analyze spatial information, edit data, maps, and present the results of all these operations. Geographic information science is the science underlying the geographic concepts, applications and systems.

Applications

GIS technology can be used for: earth surface based scientific investigations; resource management, reference, and projections of a geospatial nature—both manmade and natural; asset management and location planning; archaeology; environmental impact study; infrastructure assessment and development; urban planning; cartography, for a thematic and/or time based purpose; criminology; geospatial intelligence; GIS data development geographic history; marketing; logistics; population and demographic studies; prospectivity mapping; location attributes applied statistical analysis; warfare assessments; and other purposes. Examples of use are: GIS may allow emergency planners to easily calculate emergency response times and the movement of response resources (for logistics) in the case of a natural disaster; GIS might be used to find wetlands that need protection strategies regarding pollution; or GIS can be used by a company to site a new business location to take advantage of GIS data identified trends to respond to a previously under-served market. Most city and transportation systems planning offices have GIS sections.

History of development

In 1854, John Snow depicted a cholera outbreak in London using points to represent the locations of some individual cases, possibly the earliest use of the geographic method. His study of the distribution of cholera led to the source of the disease, a contaminated water pump (the Broad Street Pump, whose handle he had disconnected, thus terminating the outbreak) within the heart of the cholera outbreak.

E. W. Gilbert's version (1958) of John Snow's 1855 map of the Soho cholera outbreak showing the clusters of cholera cases in the London epidemic of 1854

While the basic elements of topography and theme existed previously in cartography, the John Snow map was unique, using cartographic methods not only to depict but also to analyze clusters of geographically dependent phenomena for the first time.

The early 20th century saw the development of photolithography, by which maps were separated into layers. Computer hardware development spurred by nuclear weapon research led to general-purpose computer "mapping" applications by the early 1960s.

The year 1960 saw the development of the world's first true operational GIS in Ottawa, Ontario, Canada by the federal Department of Forestry and Rural Development. Developed by Dr. Roger Tomlinson, it was called the "Canada Geographic Information System" (CGIS) and was used to store, analyze, and manipulate data collected for the

Canada Land Inventory (CLI) – an effort to determine the land capability for rural Canada by mapping information about soils, agriculture, recreation, wildlife, waterfowl, forestry, and land use at a scale of 1:50,000. A rating classification factor was also added to permit analysis.

CGIS was the world's first such system and an improvement over "mapping" applications as it provided capabilities for overlay, measurement, and digitizing/scanning. It supported a national coordinate system that spanned the continent, coded lines as "arcs" having a true embedded topology, and it stored the attribute and locational information in separate files. As a result of this, Tomlinson has become known as the "father of GIS," particularly for his use of overlays in promoting the spatial analysis of convergent geographic data. CGIS lasted into the 1990s and built a large digital land resource database in Canada. It was developed as a mainframe based system in support of federal and provincial resource planning and management. Its strength was continent-wide analysis of complex datasets. The CGIS was never available in a commercial form.

In 1964, Howard T Fisher formed the Laboratory for Computer Graphics and Spatial Analysis at the Harvard Graduate School of Design (LCGSA 1965-1991), where a number of important theoretical concepts in spatial data handling were developed, and which by the 1970s had distributed seminal software code and systems, such as 'SYMAP', 'GRID', and 'ODYSSEY' -- which served as literal and inspirational sources for subsequent commercial development—to universities, research centers, and corporations worldwide.

By the early 1980s, M&S Computing (later Intergraph), Environmental Systems Research Institute (ESRI), CARIS (Computer Aided Resource Information System) and ERDAS emerged as commercial vendors of GIS software, successfully incorporating many of the CGIS features, combining the first generation approach to separation of spatial and attribute information with a second generation approach to organizing attribute data into database structures. In parallel, the development of two public domain systems began in the late 1970s and early 1980s. MOSS, the Map Overlay and Statistical System project started in 1977 in Fort Collins, Colorado under the auspices of the Western Energy and Land Use Team (WELUT) and the U.S. Fish and Wildlife Service. GRASS GIS was begun in 1982 by the U.S. Army Corps of Engineering Research Laboratory (USA-CERL) in Champaign, Illinois, a branch of the U.S. Army Corps of Engineers to meet the need of the U.S. military for software for land management and environmental planning. The later 1980s and 1990s industry growth were spurred on by the growing use of GIS on Unix workstations and the personal computer. By the end of the 20th century, the rapid growth in various systems had been consolidated and standardized on relatively few platforms, and users were beginning to export the concept of viewing GIS data over the Internet, requiring data format and transfer standards. More recently, a growing number of free, open source GIS packages run on a range of operating systems and can be customized to perform specific tasks. Increasingly geospatial data and mapping applications are being made available via the world wide web.

Several authoritative books on the history of GIS have been published.

GIS techniques and technology

Modern GIS technologies use digital information, for which various digitized data creation methods are used. The most common method of data creation is digitization, where a hard copy map or survey plan is transferred into a digital medium through the use of a computer-aided design (CAD) program, and geo-referencing capabilities. With the wide availability of ortho-rectified imagery (both from satellite and aerial sources), heads-up digitizing is becoming the main avenue through which geographic data is extracted. Heads-up digitizing involves the tracing of geographic data directly on top of the aerial imagery instead of by the traditional method of tracing the geographic form on a separate digitizing tablet (heads-down digitizing).

Relating information from different source

GIS uses spatio-temporal (space-time) location as the key index variable for all other information. Just as a relational database containing text or numbers can relate many different tables using common key index variables, GIS can relate otherwise unrelated information by using location as the key index variable. The key is the location and/or extent in space-time.

Any variable that can be located spatially, and increasingly also temporally, can be referenced using a GIS. Locations or extents in Earth space-time may be recorded as dates/times of occurrence, and x, y, and z coordinates representing, longitude, latitude, and elevation, respectively. These GIS coordinates may represent other quantified systems of temporo-spatial reference (for example, film frame number, stream gage station, highway mile marker, surveyor benchmark, building address, street intersection, entrance gate, water depth sounding, POS or CAD drawing origin/units). Units applied to recorded temporal-spatial data can vary widely (even when using exactly the same data, see map projections), but all Earth-based spatial-temporal location and extent references should, ideally, be relatable to one another and ultimately to a "real" physical location or extent in space-time.

Related by accurate spatial information, an incredible variety of real-world and projected past or future data can be analyzed, interpreted and represented to facilitate education and decision making. This key characteristic of GIS has begun to open new avenues of scientific inquiry into behaviors and patterns of previously considered unrelated real-world information.

GIS Uncertainties

GIS accuracy depends upon source data, and how it is encoded to be data referenced. Land Surveyors have been able to provide a high level of positional accuracy utilizing the GPS derived positions. [Retrieved from Federal Geographic Data Committee] the high-resolution digital terrain and aerial imagery, [Retrieved NJGIN] the powerful computers, Web technology, are changing the quality, utility, and expectations of GIS to serve society on a grand scale, but nevertheless there are other source data that has an impact on the overall GIS accuracy like: paper maps that are not found to be very suitable to achieve the desired accuracy since the aging of maps affects their dimensional stability.

In developing a Digital Topographic Data Base for a GIS, topographical maps are the main source of data. Aerial photography and satellite images are extra sources for collecting data and identifying attributes which can be mapped in layers over a location facsimile of scale. The scale of a map and geographical rendering area representation type are very important aspects since the information content depends mainly on the scale set and resulting locatability of the map's representations. In order to digitize a map, the map has to be checked within theoretical dimensions, then scanned into a raster format, and resulting raster data has to be given a theoretical dimension by a rubber sheeting/warping technology process.

Uncertainty is a significant problem in designing a GIS because spatial data tend to be used for purposes for which they were never intended. Some maps were made many decades ago, where at that time the computer industry was not even in its perspective establishments. This has led to historical reference maps without common norms. Map accuracy is a relative issue of minor importance in cartography. All maps are established for communication ends. Maps use a historically constrained technology of pen and paper to communicate a view of the world to their users. Cartographers feel little need to communicate information based on accuracy, for when the same map is digitized and input into a GIS, the mode of use often changes. The new uses extend well beyond a determined domain for which the original map was intended and designed.

A quantitative analysis of maps brings accuracy issues into focus. The electronic and other equipment used to make measurements for GIS is far more precise than the machines of conventional map analysis. [Retrieved USGS]. The truth is that all geographical data are inherently inaccurate, and these inaccuracies will propagate through GIS operations in ways that are difficult to predict, yet have goals of conveyance in mind for original design. Accuracy Standards for 1:24000 Scales Map: 1:24,000 ± 40.00 feet

This means that when we see a point or attribute on a map, its "probable" location is within a +/- 40 foot area of its rendered reference, according to area representations and scale.

A GIS can also convert existing digital information, which may not yet be in map form, into forms it can recognize, employ for its data analysis processes, and use in forming mapping output. For example, digital satellite images generated through remote sensing can be analyzed to produce a map-like layer of digital information about vegetative covers on land locations. Another fairly recently developed resource for naming GIS location objects is the Getty Thesaurus of Geographic Names (GTGN), which is a structured vocabulary containing about 1,000,000 names and other information about places.

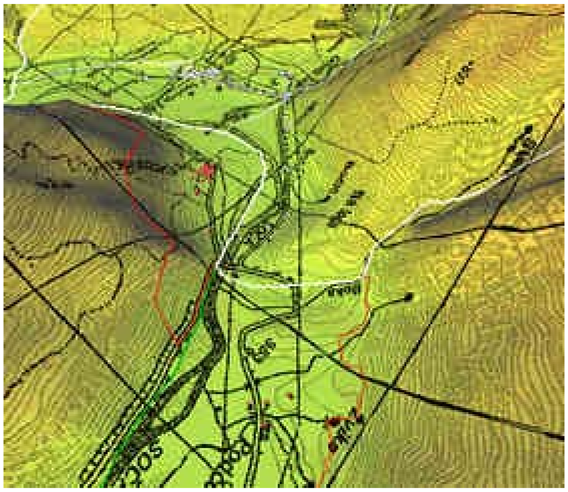
Likewise, researched census or hydrological tabular data can be displayed in map-like form, serving as layers of thematic information for forming a GIS map.

Data representation

GIS data represents real objects (such as roads, land use, elevation, trees, waterways, etc.) with digital data determining the mix. Real objects can be divided into two abstractions: discrete objects (e.g., a house) and continuous fields (such as rainfall amount, or elevations). Traditionally, there are two broad methods used to store data in a GIS for both kinds of abstractions mapping references: raster images and vector. Points, lines, and polygons are the stuff of mapped location attribute references. A new hybrid method of storing data is that of identifying point clouds, which combine three-dimensional points with RGB information at each point, returning a "3D color image". GIS Thematic maps then are becoming more and more realistically visually descriptive of what they set out to show or determine.

Raster

A raster data type is, in essence, any type of digital image represented by reducible and enlargeable grids. Anyone who is familiar with digital photography will recognize the Raster graphics pixel as the smallest individual grid unit building block of an image, usually not readily identified as an artifact shape until an image is produced on a very large scale. A combination of the pixels making up an image color formation scheme will compose details of an image, as is distinct from the commonly used points, lines, and polygon area location symbols of scalable vector graphics as the basis of the vector model of area attribute rendering. While a digital image is concerned with its output blending together its grid based details as an identifiable representation of reality, in a photograph or art image transferred into a computer, the raster data type will reflect a digitized abstraction of reality dealt with by grid populating tones or objects, quantities, cojoined or open boundaries, and map relief schemas. Aerial photos are one commonly used form of raster data, with one primary purpose in mind: to display a detailed image on a map area, or for the purposes of rendering its identifiable objects by digitization. Additional raster data sets used by a GIS will contain information regarding elevation, a digital elevation model, or reflectance of a particular wavelength of light, Landsat, or other electromagnetic spectrum indicators.



Digital elevation model, map (image), and vector data

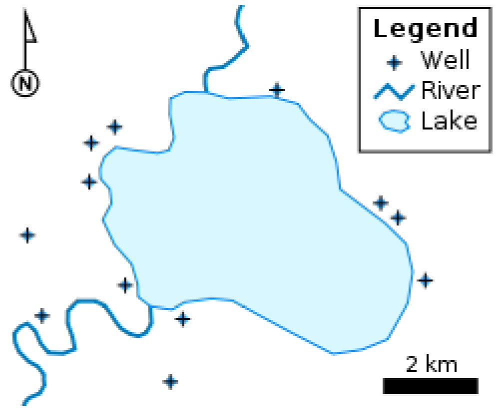
Raster data type consists of rows and columns of cells, with each cell storing a single value. Raster data can be images (raster images) with each pixel (or cell) containing a color value. Additional values recorded for each cell may be a discrete value, such as land use, a continuous value, such as temperature, or a null value if no data is available. While a raster cell stores a single value, it can be extended by using raster bands to represent RGB (red, green, blue) colors, colormaps (a mapping between a thematic code and RGB value), or an extended attribute table with one row for each unique cell value. The resolution of the raster data set is its cell width in ground units.

Raster data is stored in various formats; from a standard file-based structure of TIF, JPEG, etc. to binary large object (BLOB) data stored directly in a relational database management system (RDBMS) similar to other vector-based feature classes. Database storage, when properly indexed, typically allows for quicker retrieval of the raster data but can require storage of millions of significantly sized records.

Vector

In a GIS, geographical features are often expressed as vectors, by considering those features as geometrical shapes. Different geographical features are expressed by different types of geometry:

- Points



A simple vector map, using each of the vector elements: points for wells, lines for rivers, and a polygon for the lake.

Zero-dimensional points are used for geographical features that can best be expressed by a single point reference — in other words, by simple location. Examples include wells, peaks, features of interest, and trailheads. Points convey the least amount of information of these file types. Points can also be used to represent areas when displayed at a small scale. For example, cities on a map of the world might be represented by points rather than polygons. No measurements are possible with point features.

- Lines or polylines

One-dimensional lines or polylines are used for linear features such as rivers, roads, railroads, trails, and topographic lines. Again, as with point features, linear features displayed at a small scale will be represented as linear features rather than as a polygon. Line features can measure distance.

- Polygons

Two-dimensional polygons are used for geographical features that cover a particular area of the earth's surface. Such features may include lakes, park boundaries, buildings, city boundaries, or land uses. Polygons convey the most amount of information of the file types. Polygon features can measure perimeter and area.

Each of these geometries are linked to a row in a database that describes their attributes. For example, a database that describes lakes may contain a lake's depth, water quality, pollution level. This information can be used to make a map to describe a particular attribute of the dataset. For example, lakes could be coloured depending on level of pollution. Different geometries can also be compared. For example, the GIS could be used to identify all wells (point geometry) that are within one kilometre of a lake (polygon geometry) that has a high level of pollution.

Vector features can be made to respect spatial integrity through the application of topology rules such as 'polygons must not overlap'. Vector data can also be used to represent continuously varying phenomena. Contour lines and triangulated irregular networks (TIN) are used to represent elevation or other continuously changing values. TINs record values at point locations, which are connected by lines to form an irregular mesh of triangles. The face of the triangles represent the terrain surface.

Advantages and disadvantages

There are some important advantages and disadvantages to using a raster or vector data model to represent reality:

- Raster datasets record a value for all points in the area covered which may require more storage space than representing data in a vector format that can store data only where needed.
- Raster data allows easy implementation of overlay operations, which are more difficult with vector data.
- Vector data can be displayed as vector graphics used on traditional maps, whereas raster data will appear as an image that may have a blocky appearance for object boundaries. (depending on the resolution of the raster file)
- Vector data can be easier to register, scale, and re-project, which can simplify combining vector layers from different sources.
- Vector data is more compatible with relational database environments, where they can be part of a relational table as a normal column and processed using a multitude of operators.
- Vector file sizes are usually smaller than raster data, which can be 10 to 100 times larger than vector data (depending on resolution).
- Vector data is simpler to update and maintain, whereas a raster image will have to be completely reproduced. (Example: a new road is added).
- Vector data allows much more analysis capability, especially for "networks" such as roads, power, rail, telecommunications, etc. (Examples: Best route, largest port, airfields connected to two-lane highways). Raster data will not have all the characteristics of the features it displays.

Non-spatial data

Additional non-spatial data can also be stored along with the spatial data represented by the coordinates of a vector geometry or the position of a raster cell. In vector data, the additional data contains attributes of the feature. For example, a forest inventory polygon may also have an identifier value and information about tree species. In raster data the cell value can store attribute information, but it can also be used as an identifier that can relate to records in another table.

Software is currently being developed to support spatial and non-spatial decision-making, with the solutions to spatial problems being integrated with solutions to non-spatial problems. The end result with these Flexible Spatial Decision-Making Support Systems

(FSDSS) is expected to be that non-experts will be able to use GIS, along with spatial criteria, and simply integrate their non-spatial criteria to view solutions to multi-criteria problems. This system is intended to assist decision-making.

Data capture

Example of hardware for mapping (GPS and laser rangefinder) and data collection (rugged computer). Field GIS are current trend, accurate mapping and data analysis are done directly in the field. Presented hardware (Field-Map technology) is used mainly for forest inventories, monitoring and mapping.

Data capture—entering information into the system—consumes much of the time of GIS practitioners. There are a variety of methods used to enter data into a GIS where it is stored in a digital format.

Existing data printed on paper or PET film maps can be digitized or scanned to produce digital data. A digitizer produces vector data as an operator traces points, lines, and polygon boundaries from a map. Scanning a map results in raster data that could be further processed to produce vector data.

Survey data can be directly entered into a GIS from digital data collection systems on survey instruments using a technique called Coordinate Geometry (COGO). Positions from a Global Navigation Satellite System (GNSS) like Global Positioning System (GPS), another survey tool, can also be directly entered into a GIS. Current trend is data collection and field mapping carried out directly with field computers (position from GPS and/or laser rangefinder). New technologies allow to create maps as well as analysis directly in the field, projects are more efficient and mapping is more accurate.

Remotely sensed data also plays an important role in data collection and consist of sensors attached to a platform. Sensors include cameras, digital scanners and LIDAR, while platforms usually consist of aircraft and satellites.

The majority of digital data currently comes from photo interpretation of aerial photographs. Soft copy workstations are used to digitize features directly from stereo pairs of digital photographs. These systems allow data to be captured in two and three dimensions, with elevations measured directly from a stereo pair using principles of photogrammetry. Currently, analog aerial photos are scanned before being entered into a soft copy system, but as high quality digital cameras become cheaper this step will be skipped.

Satellite remote sensing provides another important source of spatial data. Here satellites use different sensor packages to passively measure the reflectance from parts of the electromagnetic spectrum or radio waves that were sent out from an active sensor such as radar. Remote sensing collects raster data that can be further processed using different bands to identify objects and classes of interest, such as land cover.

When data is captured, the user should consider if the data should be captured with either a relative accuracy or absolute accuracy, since this could not only influence how information will be interpreted but also the cost of data capture.

In addition to collecting and entering spatial data, attribute data is also entered into a GIS. For vector data, this includes additional information about the objects represented in the system.

After entering data into a GIS, the data usually requires editing, to remove errors, or further processing. For vector data it must be made "topologically correct" before it can be used for some advanced analysis. For example, in a road network, lines must connect with nodes at an intersection. Errors such as undershoots and overshoots must also be removed. For scanned maps, blemishes on the source map may need to be removed from the resulting raster. For example, a fleck of dirt might connect two lines that should not be connected.

Raster-to-vector translation

Data restructuring can be performed by a GIS to convert data into different formats. For example, a GIS may be used to convert a satellite image map to a vector structure by generating lines around all cells with the same classification, while determining the cell spatial relationships, such as adjacency or inclusion.

More advanced data processing can occur with image processing, a technique developed in the late 1960s by NASA and the private sector to provide contrast enhancement, false colour rendering and a variety of other techniques including use of two dimensional Fourier transforms.

Since digital data is collected and stored in various ways, the two data sources may not be entirely compatible. So a GIS must be able to convert geographic data from one structure to another.

Projections, coordinate systems and registration

A property ownership map and a soils map might show data at different scales. Map information in a GIS must be manipulated so that it registers, or fits, with information gathered from other maps. Before the digital data can be analyzed, they may have to undergo other manipulations—projection and coordinate conversions, for example—that integrate them into a GIS.

The earth can be represented by various models, each of which may provide a different set of coordinates (e.g., latitude, longitude, elevation) for any given point on the Earth's surface. The simplest model is to assume the earth is a perfect sphere. As more measurements of the earth have accumulated, the models of the earth have become more sophisticated and more accurate. In fact, there are models that apply to different areas of the earth to provide increased accuracy (e.g., North American Datum, 1927 - NAD27 - works well in North America, but not in Europe).

Projection is a fundamental component of map making. A projection is a mathematical means of transferring information from a model of the Earth, which represents a three-dimensional curved surface, to a two-dimensional medium—paper or a computer screen. Different projections are used for different types of maps because each projection particularly suits specific uses. For example, a projection that accurately represents the shapes of the continents will distort their relative sizes.

Since much of the information in a GIS comes from existing maps, a GIS uses the processing power of the computer to transform digital information, gathered from sources with different projections and/or different coordinate systems, to a common projection and coordinate system. For images, this process is called rectification.

Spatial analysis with GIS

Given the vast range of spatial analysis techniques that have been developed over the past half century, any summary or review can only cover the subject to a limited depth. This is a rapidly changing field, and GIS packages are increasingly including analytical tools as standard built-in facilities or as optional toolsets, add-ins or 'analysts'. In many instances such facilities are provided by the original software suppliers (commercial vendors or collaborative non commercial development teams), whilst in other cases facilities have been developed and are provided by third parties. Furthermore, many products offer software development kits (SDKs), programming languages and language support, scripting facilities and/or special interfaces for developing one's own analytical tools or variants. The website Geospatial Analysis and associated book/ebook attempt to provide a reasonably comprehensive guide to the subject. The impact of these myriad paths to perform spatial analysis create a new dimension to business intelligence termed "spatial intelligence" which, when delivered via intranet, democratizes access to operational sorts not usually privy to this type of information.

Slope and Aspect

Slope, aspect and surface curvature in terrain analysis are all derived from neighbourhood operations using elevation values of a cell's adjacent neighbours. Authors such as Skidmore, Jones and Zhou and Liu have compared techniques for calculating slope and aspect. Slope is a function of resolution, and the spatial resolution used to calculate slope and aspect should always be specified

The elevation at a point will have perpendicular tangents (slope) passing through the point, in an east-west and north-south direction. These two tangents give two components, $\partial z/\partial x$ and $\partial z/\partial y$, which then be used to determine the overall direction of slope, and the aspect of the slope. The gradient is defined as a vector quantity with components equal to the partial derivatives of the surface in the x and y directions.

The calculation of the overall 3x3 grid slope and aspect for methods that determine east-west and north-south component use the following formulas respectively:

$$\tan S = \sqrt{\left(\frac{\partial z}{\partial x}\right)^2 + \left(\frac{\partial z}{\partial y}\right)^2}$$

$$\tan A = \left(\frac{\left(\frac{-\partial z}{\partial y}\right)}{\left(\frac{\partial z}{\partial x}\right)}\right)$$

Zhou and Liu describe another algorithm for calculating aspect, as follows:

$$A = 270^\circ + \arctan\left(\frac{\left(\frac{\partial z}{\partial x}\right)}{\left(\frac{\partial z}{\partial y}\right)}\right) - 90^\circ\left(\frac{\left(\frac{\partial z}{\partial y}\right)}{\left|\frac{\partial z}{\partial y}\right|}\right)$$

Data modeling

It is difficult to relate wetlands maps to rainfall amounts recorded at different points such as airports, television stations, and high schools. A GIS, however, can be used to depict two- and three-dimensional characteristics of the Earth's surface, subsurface, and atmosphere from information points. For example, a GIS can quickly generate a map with isopleth or contour lines that indicate differing amounts of rainfall.

Such a map can be thought of as a rainfall contour map. Many sophisticated methods can estimate the characteristics of surfaces from a limited number of point measurements. A two-dimensional contour map created from the surface modeling of rainfall point measurements may be overlaid and analyzed with any other map in a GIS covering the same area.

Additionally, from a series of three-dimensional points, or digital elevation model, isopleth lines representing elevation contours can be generated, along with slope analysis, shaded relief, and other elevation products. Watersheds can be easily defined for any given reach, by computing all of the areas contiguous and uphill from any given point of interest. Similarly, an expected thalweg of where surface water would want to travel in intermittent and permanent streams can be computed from elevation data in the GIS.

Topological modeling

A GIS can recognize and analyze the spatial relationships that exist within digitally stored spatial data. These topological relationships allow complex spatial modelling and analysis to be performed. Topological relationships between geometric entities traditionally include adjacency (what adjoins what), containment (what encloses what), and proximity (how close something is to something else).

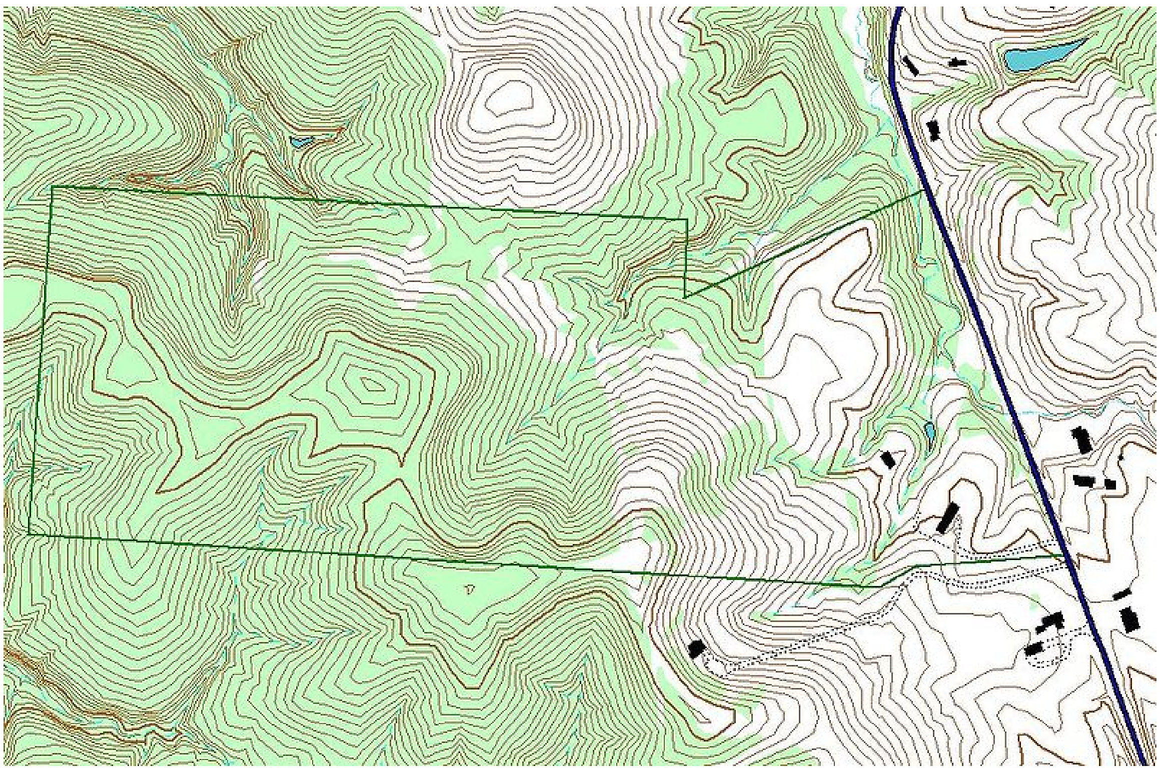
Networks

If all the factories near a wetland were accidentally to release chemicals into the river at the same time, how long would it take for a damaging amount of pollutant to enter the wetland reserve? A GIS can simulate the routing of materials along a linear network. Values such as slope, speed limit, or pipe diameter can be incorporated into network modeling to represent the flow of the phenomenon more accurately. Network modelling is commonly employed in transportation planning, hydrology modeling, and infrastructure modeling.

Hydrological Modeling

GIS hydrological models can provide a spatial element that other hydrological models lack, with the analysis of variables such as slope, aspect and watershed or catchment area. Terrain analysis is fundamental to hydrology, since water always flows down a slope. As basic terrain analysis of a DEM involves calculation of slope and aspect, DEMs are very useful for hydrological analysis. Slope and aspect can then be used to determine direction of surface runoff, and hence flow accumulation for the formation of streams, rivers and lakes. Areas of divergent flow can also give a clear indication of the boundaries of a catchment. Once a flow direction and accumulation matrix has been created, queries can be performed that show contributing or dispersal areas at a certain point. More detail can be added to the model, such as terrain roughness, vegetation types and soil types, which can influence infiltration and evapotranspiration rates, and hence influencing surface flow. These extra layers of detail ensures a more accurate model.

Cartographic modeling



An example of use of layers in a GIS application. In this example, the forest cover layer (light green) is at the bottom, with the topographic layer over it. Next up is the stream layer, then the boundary layer, then the road layer. The order is very important in order to properly display the final result. Note that the pond layer was located just below the stream layer, so that a stream line can be seen overlying one of the ponds.

The term "cartographic modeling" was (probably) coined by Dana Tomlin in his PhD dissertation and later in his book which has the term in the title. Cartographic modeling refers to a process where several thematic layers of the same area are produced, processed, and analyzed. Tomlin used raster layers, but the overlay method (see below) can be used more generally. Operations on map layers can be combined into algorithms, and eventually into simulation or optimization models.

Map overlay

The combination of several spatial datasets (points, lines or polygons) creates a new output vector dataset, visually similar to stacking several maps of the same region. These overlays are similar to mathematical Venn diagram overlays. A union overlay combines the geographic features and attribute tables of both inputs into a single new output. An intersect overlay defines the area where both inputs overlap and retains a set of attribute fields for each. A symmetric difference overlay defines an output area that includes the total area of both inputs except for the overlapping area.

Data extraction is a GIS process similar to vector overlay, though it can be used in either vector or raster data analysis. Rather than combining the properties and features of both datasets, data extraction involves using a "clip" or "mask" to extract the features of one data set that fall within the spatial extent of another dataset.

In raster data analysis, the overlay of datasets is accomplished through a process known as "local operation on multiple rasters" or "map algebra," through a function that combines the values of each raster's matrix. This function may weigh some inputs more than others through use of an "index model" that reflects the influence of various factors upon a geographic phenomenon.

Automated cartography

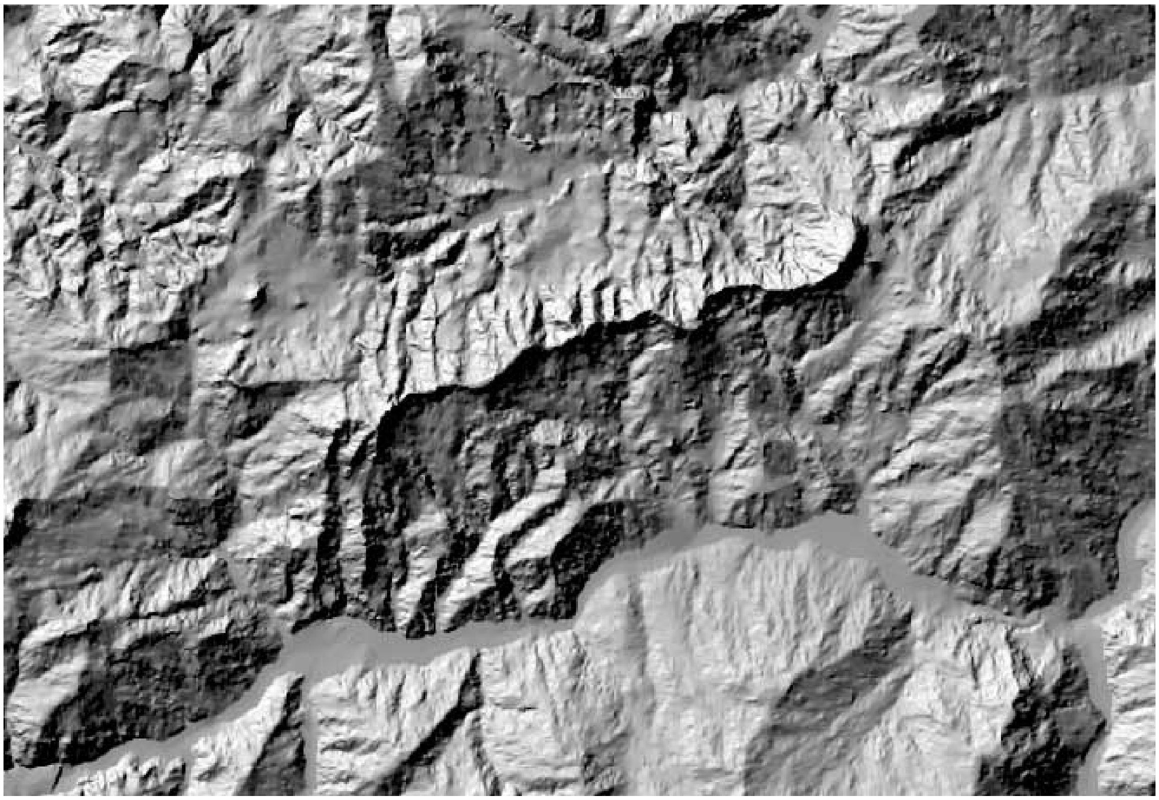
Digital cartography and GIS both encode spatial relationships in structured formal representations. GIS is used in digital cartography modeling as a (semi)automated process of making maps, so called Automated Cartography. In practice, it can be a subset of a GIS, within which it is equivalent to the stage of visualization, since in most cases not all of the GIS functionality is used. Cartographic products can be either in a digital or in a hardcopy format. Powerful analysis techniques with different data representation can produce high-quality maps within a short time period. The main problem in Automated Cartography is to use a single set of data to produce multiple products at a variety of scales, a technique known as cartographic generalization.

Geostatistics

Geostatistics is a point-pattern analysis that produces field predictions from data points. It is a way of looking at the statistical properties of those special data. It is different from general applications of statistics because it employs the use of graph theory and matrix algebra to reduce the number of parameters in the data. Only the second-order properties of the GIS data are analyzed.

When phenomena are measured, the observation methods dictate the accuracy of any subsequent analysis. Due to the nature of the data (e.g. traffic patterns in an urban environment; weather patterns over the Pacific Ocean), a constant or dynamic degree of precision is always lost in the measurement. This loss of precision is determined from the scale and distribution of the data collection.

To determine the statistical relevance of the analysis, an average is determined so that points (gradients) outside of any immediate measurement can be included to determine their predicted behavior. This is due to the limitations of the applied statistic and data collection methods, and interpolation is required to predict the behavior of particles, points, and locations that are not directly measurable.



Hillshade model derived from a Digital Elevation Model (DEM) of the Valestra area in the northern Apennines (Italy)

Interpolation is the process by which a surface is created, usually a raster dataset, through the input of data collected at a number of sample points. There are several forms of interpolation, each which treats the data differently, depending on the properties of the data set. In comparing interpolation methods, the first consideration should be whether or not the source data will change (exact or approximate). Next is whether the method is subjective, a human interpretation, or objective. Then there is the nature of transitions between points: are they abrupt or gradual. Finally, there is whether a method is global (it uses the entire data set to form the model), or local where an algorithm is repeated for a small section of terrain.

Interpolation is a justified measurement because of a spatial autocorrelation principle that recognizes that data collected at any position will have a great similarity to, or influence of those locations within its immediate vicinity.

Digital elevation models (DEM), triangulated irregular networks (TIN), edge finding algorithms, Thiessen polygons, Fourier analysis, (weighted) moving averages, inverse distance weighting, kriging, spline, and trend surface analysis are all mathematical methods to produce interpolative data.

Address geocoding

Geocoding is interpolating spatial locations (X,Y coordinates) from street addresses or any other spatially referenced data such as ZIP Codes, parcel lots and address locations. A reference theme is required to geocode individual addresses, such as a road centerline file with address ranges. The individual address locations have historically been interpolated, or estimated, by examining address ranges along a road segment. These are usually provided in the form of a table or database. The GIS will then place a dot approximately where that address belongs along the segment of centerline. For example, an address point of 500 will be at the midpoint of a line segment that starts with address 1 and ends with address 1000. Geocoding can also be applied against actual parcel data, typically from municipal tax maps. In this case, the result of the geocoding will be an actually positioned space as opposed to an interpolated point. This approach is being increasingly used to provide more precise location information.

There are several potentially dangerous caveats that are often overlooked when using interpolation.

Various algorithms are used to help with address matching when the spellings of addresses differ. Address information that a particular entity or organization has data on, such as the post office, may not entirely match the reference theme. There could be variations in street name spelling, community name, etc. Consequently, the user generally has the ability to make matching criteria more stringent, or to relax those parameters so that more addresses will be mapped. Care must be taken to review the results so as not to map addresses incorrectly due to overzealous matching parameters.

Reverse geocoding

Reverse geocoding is the process of returning an estimated street address number as it relates to a given coordinate. For example, a user can click on a road centerline theme (thus providing a coordinate) and have information returned that reflects the estimated house number. This house number is interpolated from a range assigned to that road segment. If the user clicks at the midpoint of a segment that starts with address 1 and ends with 100, the returned value will be somewhere near 50. Note that reverse geocoding does not return actual addresses, only estimates of what should be there based on the predetermined range.

Data output and cartography

Cartography is the design and production of maps, or visual representations of spatial data. The vast majority of modern cartography is done with the help of computers, usually using a GIS but production quality cartography is also achieved by importing layers into a design program to refine it. Most GIS software gives the user substantial control over the appearance of the data.

Cartographic work serves two major functions:

First, it produces graphics on the screen or on paper that convey the results of analysis to the people who make decisions about resources. Wall maps and other graphics can be generated, allowing the viewer to visualize and thereby understand the results of analyses or simulations of potential events. Web Map Servers facilitate distribution of generated maps through web browsers using various implementations of web-based application programming interfaces (AJAX, Java, Flash, etc.).

Second, other database information can be generated for further analysis or use. An example would be a list of all addresses within one mile (1.6 km) of a toxic spill.

Graphic display techniques

Traditional maps are abstractions of the real world, a sampling of important elements portrayed on a sheet of paper with symbols to represent physical objects. People who use maps must interpret these symbols. Topographic maps show the shape of land surface with contour lines or with shaded relief.

Today, graphic display techniques such as shading based on altitude in a GIS can make relationships among map elements visible, heightening one's ability to extract and analyze information. For example, two types of data were combined in a GIS to produce a perspective view of a portion of San Mateo County, California.

- The digital elevation model, consisting of surface elevations recorded on a 30-meter horizontal grid, shows high elevations as white and low elevation as black.
- The accompanying Landsat Thematic Mapper image shows a false-color infrared image looking down at the same area in 30-meter pixels, or picture elements, for the same coordinate points, pixel by pixel, as the elevation information.

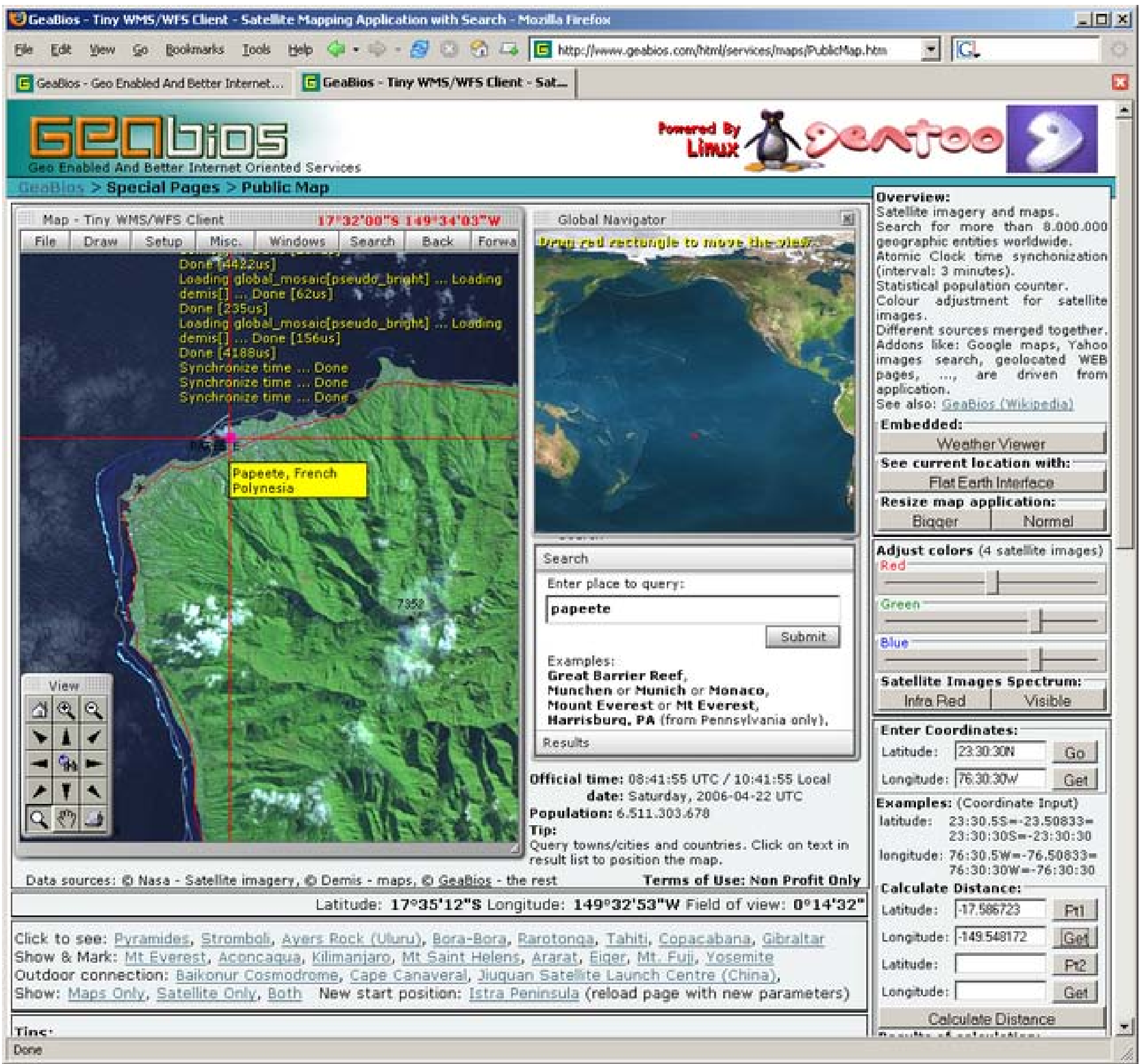
A GIS was used to register and combine the two images to render the three-dimensional perspective view looking down the San Andreas Fault, using the Thematic Mapper image pixels, but shaded using the elevation of the landforms. The GIS display depends on the viewing point of the observer and time of day of the display, to properly render the shadows created by the sun's rays at that latitude, longitude, and time of day.

An archeochrome is a new way of displaying spatial data. It is a thematic on a 3D map that is applied to a specific building or a part of a building. It is suited to the visual display of heat loss data.

Spatial ETL

Spatial ETL tools provide the data processing functionality of traditional Extract, Transform, Load (ETL) software, but with a primary focus on the ability to manage spatial data. They provide GIS users with the ability to translate data between different standards and proprietary formats, whilst geometrically transforming the data en-route.

GIS developments



GeaBios - tiny WMS/WFS client (Flash/DHTML)

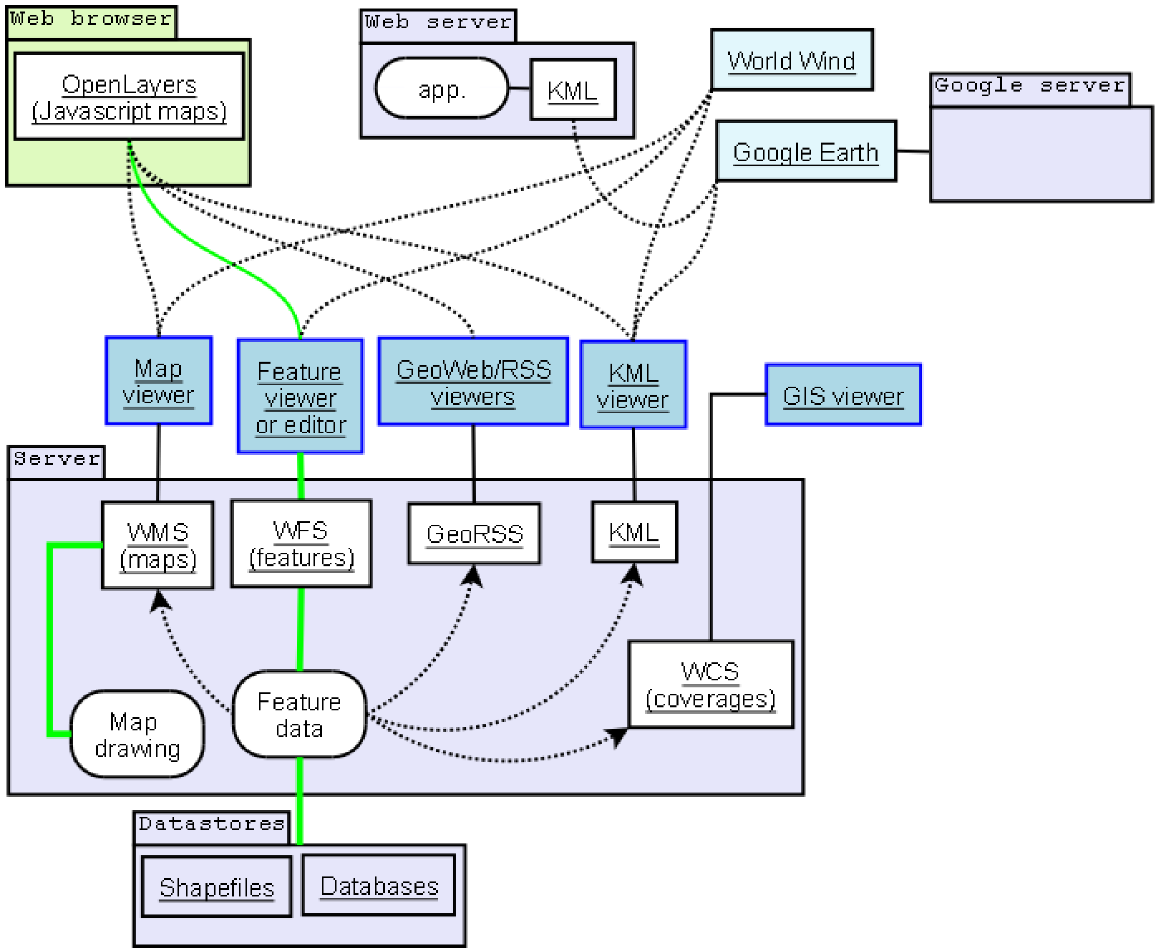
Many disciplines can benefit from GIS technology. An active GIS market has resulted in lower costs and continual improvements in the hardware and software components of

GIS. These developments will, in turn, result in a much wider use of the technology throughout science, government, business, and industry, with applications including real estate, public health, crime mapping, national defense, sustainable development, natural resources, landscape architecture, archaeology, regional and community planning, transportation and logistics. GIS is also diverging into location-based services (LBS). LBS allows GPS enabled mobile devices to display their location in relation to fixed assets (nearest restaurant, gas station, fire hydrant), mobile assets (friends, children, police car) or to relay their position back to a central server for display or other processing. These services continue to develop with the increased integration of GPS functionality with increasingly powerful mobile electronics (cell phones, PDAs, laptops).

OGC standards

The Open Geospatial Consortium (OGC) is an international industry consortium of 384 companies, government agencies, universities and individuals participating in a consensus process to develop publicly available geoprocessing specifications. Open interfaces and protocols defined by OpenGIS Specifications support interoperable solutions that "geo-enable" the Web, wireless and location-based services, and mainstream IT, and empower technology developers to make complex spatial information and services accessible and useful with all kinds of applications. Open Geospatial Consortium (OGC) protocols include Web Map Service (WMS) and Web Feature Service (WFS).

GIS products are broken down by the OGC into two categories, based on how completely and accurately the software follows the OGC specifications.



OGC standards help GIS tools communicate.

Compliant Products are software products that comply to OGC's OpenGIS Specifications. When a product has been tested and certified as compliant through the OGC Testing Program, the product is automatically registered as "compliant" on this site.

Implementing Products are software products that implement OpenGIS Specifications but have not yet passed a compliance test. Compliance tests are not available for all specifications. Developers can register their products as implementing draft or approved specifications, though OGC reserves the right to review and verify each entry.

Web mapping

In recent years there has been an explosion of mapping applications on the web such as Google Maps and Bing Maps. These websites give the public access to huge amounts of geographic data.

Some of them, like Google Maps and OpenLayers, expose an API that enable users to create custom applications. These toolkits commonly offer street maps, aerial/satellite imagery, geocoding, searches, and routing functionality.

Other applications for publishing geographic information on the web include GeoBase (Telogis GIS software), Smallworld's SIAS or GSS, MapInfo's MapXtreme or PlanAcess or Stratus Connect, Cadcorp's GeognoSIS, Intergraph's GeoMedia WebMap (TM), ESRI's ArcIMS, ArcGIS Server, Autodesk's Mapguide, SeaTrails' AtlasAlive, ObjectFX's Web Mapping Tools, ERDAS APOLLO Suite, Google Earth, Google Fusion Tables, and the open source MapServer or GeoServer.

In recent years web mapping services have begun to adopt features more common in GIS. Services such as Google Maps and Bing Maps allow users to access and annotate maps and share the maps with others.

Global change, climate history program and prediction of its impact

Maps have traditionally been used to explore the Earth and to exploit its resources. GIS technology, as an expansion of cartographic science, has enhanced the efficiency and analytic power of traditional mapping. Now, as the scientific community recognizes the environmental consequences of anthropogenic activities influencing climate change, GIS technology is becoming an essential tool to understand the impacts of this change over time. GIS enables the combination of various sources of data with existing maps and up-to-date information from earth observation satellites along with the outputs of climate change models. This can help in understanding the effects of climate change on complex natural systems. One of the classic examples of this is the study of Arctic Ice Melting.

The outputs from a GIS in the form of maps combined with satellite imagery allow researchers to view their subjects in ways that literally never have been seen before. The images are also invaluable for conveying the effects of climate change to non-scientists.

Adding the dimension of time

The condition of the Earth's surface, atmosphere, and subsurface can be examined by feeding satellite data into a GIS. GIS technology gives researchers the ability to examine the variations in Earth processes over days, months, and years.

As an example, the changes in vegetation vigor through a growing season can be animated to determine when drought was most extensive in a particular region. The resulting graphic, known as a normalized vegetation index, represents a rough measure of plant health. Working with two variables over time would then allow researchers to detect regional differences in the lag between a decline in rainfall and its effect on vegetation.

GIS technology and the availability of digital data on regional and global scales enable such analyses. The satellite sensor output used to generate a vegetation graphic is produced for example by the Advanced Very High Resolution Radiometer (AVHRR). This sensor system detects the amounts of energy reflected from the Earth's surface across various bands of the spectrum for surface areas of about 1 square kilometer. The satellite sensor produces images of a particular location on the Earth twice a day.

AVHRR and more recently the Moderate-Resolution Imaging Spectroradiometer (MODIS) are only two of many sensor systems used for Earth surface analysis. More sensors will follow, generating ever greater amounts of data.

GIS and related technology will help greatly in the management and analysis of these large volumes of data, allowing for better understanding of terrestrial processes and better management of human activities to maintain world economic vitality and environmental quality.

In addition to the integration of time in environmental studies, GIS is also being explored for its ability to track and model the progress of humans throughout their daily routines. A concrete example of progress in this area is the recent release of time-specific population data by the US Census. In this data set, the populations of cities are shown for daytime and evening hours highlighting the pattern of concentration and dispersion generated by North American commuting patterns. The manipulation and generation of data required to produce this data would not have been possible without GIS.

Using models to project the data held by a GIS forward in time have enabled planners to test policy decisions. These systems are known as Spatial Decision Support Systems.

Semantics

Tools and technologies emerging from the W3C's Semantic Web Activity are proving useful for data integration problems in information systems. Correspondingly, such technologies have been proposed as a means to facilitate interoperability and data reuse among GIS applications and also to enable new analysis mechanisms.

Ontologies are a key component of this semantic approach as they allow a formal, machine-readable specification of the concepts and relationships in a given domain. This in turn allows a GIS to focus on the intended meaning of data rather than its syntax or structure. For example, reasoning that a land cover type classified as *deciduous needleleaf trees* in one dataset is a specialization or subset of land cover type *forest* in another more roughly classified dataset can help a GIS automatically merge the two datasets under the more general land cover classification. Tentative ontologies have been developed in areas related to GIS applications, for example the hydrology ontology developed by the Ordnance Survey in the United Kingdom and the SWEET ontologies developed by NASA's Jet Propulsion Laboratory. Also, simpler ontologies and semantic metadata standards are being proposed by the W3C Geo Incubator Group to represent geospatial data on the web.

Recent research results in this area can be seen in the International Conference on Geospatial Semantics and the Terra Cognita -- Directions to the Geospatial Semantic Web workshop at the International Semantic Web Conference.

Society

With the popularization of GIS in decision making, scholars have begun to scrutinize the social implications of GIS. It has been argued that the production, distribution, utilization, and representation of geographic information are largely related with the social context. Other related topics include discussion on copyright, privacy, and censorship. A more optimistic social approach to GIS adoption is to use it as a tool for public participation.

Chapter 3

Intelligent Transportation System

The term ***intelligent transport system*** (ITS) refers to efforts to add information and communications technology to transport infrastructure and vehicles in an effort to manage factors that typically are at odds with each other, such as vehicles, loads, and routes to improve safety and reduce vehicle wear, transportation times, and fuel consumption.

Background

Interest in ITS comes from the problems caused by traffic congestion and a synergy of new information technology for simulation, real-time control, and communications networks. Traffic congestion has been increasing worldwide as a result of increased motorization, urbanization, population growth, and changes in population density. Congestion reduces efficiency of transportation infrastructure and increases travel time, air pollution, and fuel consumption.

The United States, for example, saw large increases in both motorization and urbanization starting in the 1920s that led to migration of the population from the sparsely populated rural areas and the densely packed urban areas into suburbs. The industrial economy replaced the agricultural economy, leading the population to move from rural locations into urban centers. At the same time, motorization was causing cities to expand because motorized transportation could not support the population density that the existing mass transit systems could. Suburbs provided a reasonable compromise between population density and access to a wide variety of employment, goods, and services that were available in the more densely populated urban centers. Further, suburban infrastructure could be built quickly, supporting a rapid transition from a rural/agricultural economy to an industrial/urban economy.

Recent governmental activity in the area of ITS – specifically in the United States – is further motivated by the perceived need for homeland security. Many of the proposed

ITS systems also involve surveillance of the roadways, which is a priority of homeland security. Funding of many systems comes either directly through homeland security organizations or with their approval. Further, ITS can play a role in the rapid mass evacuation of people in urban centers after large casualty events such as a result of a natural disaster or threat. Much of the infrastructure and planning involved with ITS parallels the need for homeland security systems.

In the developing world, the migration of people from rural to urbanized habitats has progressed differently. Many areas of the developing world have urbanized without significant motorization and the formation of suburbs. In areas like Santiago, Chile, a high population density is supported by a multimodal system of walking, bicycle transportation, motorcycles, buses, and trains. A small portion of the population can afford automobiles, but the automobiles greatly increase the congestion in these multimodal transportation systems. They also produce a considerable amount of air pollution, pose a significant safety risk, and exacerbate feelings of inequities in the society.

Other parts of the developing world, such as China, remain largely rural but are rapidly urbanizing and industrializing. In these areas a motorized infrastructure is being developed alongside motorization of the population. Great disparity of wealth means that only a fraction of the population can motorize, and therefore the highly dense multimodal transportation system for the poor is cross-cut by the highly motorized transportation system for the rich. The urban infrastructure is being rapidly developed, providing an opportunity to build new systems that incorporate ITS at early stages.

Intelligent transport technologies

Intelligent transport systems vary in technologies applied, from basic management systems such as car navigation; traffic signal control systems; container management systems; variable message signs; automatic number plate recognition or speed cameras to monitor applications, such as security CCTV systems; and to more advanced applications that integrate live data and feedback from a number of other sources, such as parking guidance and information systems; weather information; bridge deicing systems; and the like. Additionally, predictive techniques are being developed to allow advanced modeling and comparison with historical baseline data. Some of the constituent technologies typically implemented in ITS are described in the following sections.

Wireless communications

Various forms of wireless communications technologies have been proposed for intelligent transportation systems.

Radio modem communication on UHF and VHF frequencies are widely used for short and long range communication within ITS.

Short-range communications (less than 500 yards) can be accomplished using IEEE 802.11 protocols, specifically WAVE or the Dedicated Short Range Communications standard being promoted by the Intelligent Transportation Society of America and the United States Department of Transportation. Theoretically, the range of these protocols can be extended using Mobile ad-hoc networks or Mesh networking.

Longer range communications have been proposed using infrastructure networks such as WiMAX (IEEE 802.16), Global System for Mobile Communications (GSM), or 3G. Long-range communications using these methods are well established, but, unlike the short-range protocols, these methods require extensive and very expensive infrastructure deployment. There is lack of consensus as to what business model should support this infrastructure.

Computational technologies

Recent advances in vehicle electronics have led to a move toward fewer, more capable computer processors on a vehicle. A typical vehicle in the early 2000s would have between 20 and 100 individual networked microcontroller/Programmable logic controller modules with non-real-time operating systems. The current trend is toward fewer, more costly microprocessor modules with hardware memory management and Real-Time Operating Systems. The new embedded system platforms allow for more sophisticated software applications to be implemented, including model-based process control, artificial intelligence, and ubiquitous computing. Perhaps the most important of these for Intelligent Transportation Systems is artificial intelligence.

Floating car data/floating cellular data

"Floating car" or "probe" data collection is a set of relatively low-cost methods for obtaining travel time and speed data for vehicles traveling along streets, highways, freeways, and other transportation routes. Broadly speaking, three methods have been used to obtain the raw data:

- **Triangulation Method.** In developed countries a high proportion of cars contain one or more mobile phones. The phones periodically transmit their presence information to the mobile phone network, even when no voice connection is established. In the mid 2000s, attempts were made to use mobile phones as anonymous traffic probes. As a car moves, so does the signal of any mobile phones that are inside the vehicle. By measuring and analyzing network data using triangulation, pattern matching or cell-sector statistics (in an anonymous format), the data was converted into traffic flow information. With more congestion, there are more cars, more phones, and thus, more probes. In metropolitan areas, the distance between antennas is shorter and in theory accuracy increases. An advantage of this method is that no infrastructure needs to be built along the road; only the mobile phone network is leveraged. But in practice the triangulation method can be complicated, especially in areas where the same mobile phone towers serve two or more parallel routes (such as a

freeway with a frontage road, a freeway and a commuter rail line, two or more parallel streets, or a street that is also a bus line). By the early 2010s, the popularity of the triangulation method was declining.
- **Vehicle Re-Identification.** Vehicle re-identification methods require sets of detectors mounted along the road. In this technique, a unique serial number for a device in the vehicle is detected at one location and then detected again (re-identified) further down the road. Travel times and speed are calculated by comparing the time at which a specific device is detected by pairs of sensors. This can be done using the MAC (Machine Access Control) addresses from Bluetooth devices, or using the RFID serial numbers from Electronic Toll Collection (ETC) transponders (also called "toll tags").
- **GPS Based Methods.** An increasing number of vehicles are equipped with in-vehicle GPS (sattelite navigation) systems that have two-way communication with a traffic data provider. Position readings from these vehicles are used to compute vehicle speeds.

Floating car data technology provides advantages over other methods of traffic measurement:

- Less expensive than sensors or cameras
- More coverage (potentially including all locations and streets)
- Faster to set up and less maintenance
- Works in all weather conditions, including heavy rain

Sensing technologies

Technological advances in telecommunications and information technology, coupled with state-of-the-art microchip, RFID (Radio Frequency Identification), and inexpensive intelligent beacon sensing technologies, have enhanced the technical capabilities that will facilitate motorist safety benefits for intelligent transportation systems globally. Sensing systems for ITS are vehicle- and infrastructure-based networked systems, i.e., Intelligent vehicle technologies. Infrastructure sensors are indestructible (such as in-road reflectors) devices that are installed or embedded in the road or surrounding the road (e.g., on buildings, posts, and signs), as required, and may be manually disseminated during preventive road construction maintenance or by sensor injection machinery for rapid deployment. Vehicle-sensing systems include deployment of infrastructure-to-vehicle and vehicle-to-infrastructure electronic beacons for identification communications and may also employ video automatic number plate recognition or vehicle magnetic signature detection technologies at desired intervals to increase sustained monitoring of vehicles operating in critical zones.

Inductive loop detection

Inductive loops can be placed in a roadbed to detect vehicles as they pass through the loop's magnetic field. The simplest detectors simply count the number of vehicles during a unit of time (typically 60 seconds in the United States) that pass over the loop, while more sophisticated sensors estimate the speed, length, and weight of vehicles and the distance between them. Loops can be placed in a single lane or across multiple lanes, and they work with very slow or stopped vehicles as well as vehicles moving at high-speed.

Video vehicle detection

Traffic flow measurement and automatic incident detection using video cameras is another form of vehicle detection. Since video detection systems such as those used in automatic number plate recognition do not involve installing any components directly into the road surface or roadbed, this type of system is known as a "non-intrusive" method of traffic detection. Video from black-and-white or color cameras is fed into processors that analyze the changing characteristics of the video image as vehicles pass. The cameras are typically mounted on poles or structures above or adjacent to the roadway. Most video detection systems require some initial configuration to "teach" the processor the baseline background image. This usually involves inputting known measurements such as the distance between lane lines or the height of the camera above the roadway. A single video detection processor can detect traffic simultaneously from one to eight cameras, depending on the brand and model. The typical output from a video detection system is lane-by-lane vehicle speeds, counts, and lane occupancy readings. Some systems provide additional outputs including gap, headway, stopped-vehicle detection, and wrong-way vehicle alarms.

Intelligent transport applications

Emergency vehicle notification systems

The in-vehicle eCall is an emergency call generated either manually by the vehicle occupants or automatically via activation of in-vehicle sensors after an accident. When activated, the in-vehicle eCall device will establish an emergency call carrying both voice and data directly to the nearest emergency point (normally the nearest E1-1-2 Public-safety answering point, PSAP). The voice call enables the vehicle occupant to communicate with the trained eCall operator. At the same time, a minimum set of data will be sent to the eCall operator receiving the voice call.

The minimum set of data contains information about the incident, including time, precise location, the direction the vehicle was traveling, and vehicle identification. The pan-European eCall aims to be operative for all new type-approved vehicles as a standard option. Depending on the manufacturer of the eCall system, it could be mobile phone based (Bluetooth connection to an in-vehicle interface), an integrated eCall device, or a functionality of a broader system like navigation, Telematics device, or tolling device. eCall is expected to be offered, at earliest, by the end of 2010, pending standardization by the European Telecommunications Standards Institute and commitment from large EU member states such as France and the United Kingdom.

Congestion pricing gantry at North Bridge Road, Singapore.

Automatic road enforcement

Automatic speed enforcement gantry or "*Lombada Eletrônica*" with ground sensors at Brasilia, D.F.

A traffic enforcement camera system, consisting of a camera and a vehicle-monitoring device, is used to detect and identify vehicles disobeying a speed limit or some other road legal requirement and automatically ticket offenders based on the license plate number. Traffic tickets are sent by mail. Applications include:

- Speed cameras that identify vehicles traveling over the legal speed limit. Many such devices use radar to detect a vehicle's speed or electromagnetic loops buried in each lane of the road.
- Red light cameras that detect vehicles that cross a stop line or designated stopping place while a red traffic light is showing.
- Bus lane cameras that identify vehicles traveling in lanes reserved for buses. In some jurisdictions, bus lanes can also be used by taxis or vehicles engaged in car pooling.
- Level crossing cameras that identify vehicles crossing railways at grade illegally.
- Double white line cameras that identify vehicles crossing these lines.
- High-occupancy vehicle lane cameras for that identify vehicles violating HOV requirements.
- Turn cameras at intersections where specific turns are prohibited on red. This type of camera is mostly used in cities or heavy populated areas.

Variable speed limits

Example variable speed limit sign in the United States.

Recently some jurisdictions have begun experimenting with variable speed limits that change with road congestion and other factors. Typically such speed limits only change to decline during poor conditions, rather than being improved in good ones. One example is on Britain's M25 motorway, which circumnavigates London. On the most heavily-traveled 14-mile (23 km) section (junction 10 to 16) of the M25 variable speed limits combined with automated enforcement have been in force since 1995. Initial results indicated savings in journey times, smoother-flowing traffic, and a fall in the number of accidents, so the implementation was made permanent in 1997. Further trials on the M25 have been thus far proved inconclusive.

Collision avoidance systems

Japan has installed sensors on its highways to notify motorists that a car is stalled ahead.

Dynamic Traffic Light Sequence

Intelligent RFID traffic control has been developed for dynamic traffic light sequence. It circumvents or avoids problems that usually arise with systems that use image processing and beam interruption techniques. RFID technology with appropriate algorithm and database were applied to a multi vehicle, multi lane and multi road junction area to provide an efficient time management scheme. A dynamic time schedule was worked out for the passage of each column. The simulation has shown that, the dynamic sequence algorithm has the ability to intelligently adjust itself even with the presence of some extreme cases. The real time operation of the system able to emulate the judgment of a traffic policeman on duty, by considering the number of vehicles in each column and the routing proprieties.

Cooperative systems on the road

Communication cooperation on the road includes car-to-car, car-to-infrastructure, and vice versa. Data available from vehicles is acquired and transmitted to a server for central fusion and processing. This data can be used to detect events such as rain (wiper activity) and congestion (frequent braking activities). The server processes a driving recommendation dedicated to a single or a specific group of drivers and transmits it wirelessly to vehicles. The goal of cooperative systems is to use and plan communication and sensor infrastructure to increase road safety. The definition of cooperative systems in road traffic is according to the European Commission:

> "Road operators, infrastructure, vehicles, their drivers and other road users will cooperate to deliver the most efficient, safe, secure and comfortable journey. The vehicle-vehicle and vehicle-infrastructure co-operative systems will contribute to these objectives beyond the improvements achievable with stand-alone systems."

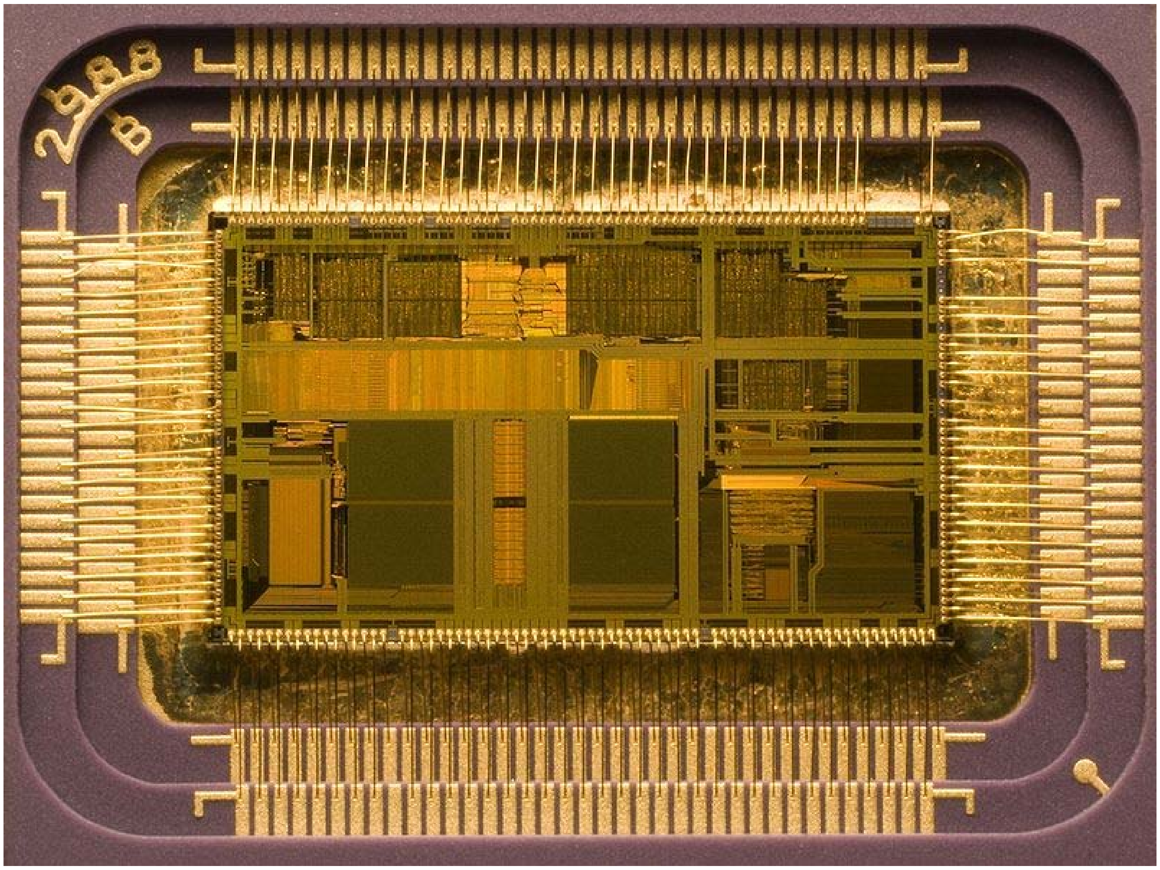
Chapter 4

Digital Electronics

Digital electronics represent signals by discrete bands of analog levels, rather than by a continuous range. All levels within a band represent the same signal state. Relatively small changes to the analog signal levels due to manufacturing tolerance, signal attenuation or parasitic noise do not leave the discrete envelope, and as a result are ignored by signal state sensing circuitry.

In most cases the number of these states is two, and they are represented by two voltage bands: one near a reference value (typically termed as "ground" or zero volts) and a value near the supply voltage, corresponding to the "false" ("0") and "true" ("1") values of the boolean domain respectively.

An industrial digital controller



Intel 80486DX2 microprocessor

Digital techniques are useful because it is easier to get an electronic device to switch into one of a number of known states than to accurately reproduce a continuous range of values.

Digital electronic circuits are usually made from large assemblies of logic gates, simple electronic representations of Boolean logic functions.

Advantages

One advantage of digital circuits when compared to analog circuits is that signals represented digitally can be transmitted without degradation due to noise. For example, a continuous audio signal, transmitted as a sequence of 1s and 0s, can be reconstructed without error provided the noise picked up in transmission is not enough to prevent identification of the 1s and 0s. An hour of music can be stored on a compact disc as about 6 billion binary digits.

In a digital system, a more precise representation of a signal can be obtained by using more binary digits to represent it. While this requires more digital circuits to process the signals, each digit is handled by the same kind of hardware. In an analog system, additional resolution requires fundamental improvements in the linearity and noise characteristics of each step of the signal chain.

Computer-controlled digital systems can be controlled by software, allowing new functions to be added without changing hardware. Often this can be done outside of the factory by updating the product's software. So, the product's design errors can be corrected after the product is in a customer's hands.

Information storage can be easier in digital systems than in analog ones. The noise-immunity of digital systems permits data to be stored and retrieved without degradation. In an analog system, noise from aging and wear degrade the information stored. In a digital system, as long as the total noise is below a certain level, the information can be recovered perfectly.

Disadvantages

In some cases, digital circuits use more energy than analog circuits to accomplish the same tasks, thus producing more heat which increases the complexity of the circuits such as the inclusion of heat sinks. In portable or battery-powered systems this can limit use of digital systems.

For example, battery-powered cellular telephones often use a low-power analog front-end to amplify and tune in the radio signals from the base station. However, a base station has grid power and can use power-hungry, but very flexible software radios. Such base stations can be easily reprogrammed to process the signals used in new cellular standards.

Digital circuits are sometimes more expensive, especially in small quantities.

Most useful digital systems must translate from continuous analog signals to discrete digital signals. This causes quantization errors. Quantization error can be reduced if the system stores enough digital data to represent the signal to the desired degree of fidelity. The Nyquist-Shannon sampling theorem provides an important guideline as to how much digital data is needed to accurately portray a given analog signal.

In some systems, if a single piece of digital data is lost or misinterpreted, the meaning of large blocks of related data can completely change. Because of the cliff effect, it can be difficult for users to tell if a particular system is right on the edge of failure, or if it can tolerate much more noise before failing.

Digital fragility can be reduced by designing a digital system for robustness. For example, a parity bit or other error management method can be inserted into the signal path. These schemes help the system detect errors, and then either correct the errors, or at least ask for a new copy of the data. In a state-machine, the state transition logic can be designed to catch unused states and trigger a reset sequence or other error recovery routine.

Digital memory and transmission systems can use techniques such as error detection and correction to use additional data to correct any errors in transmission and storage.

On the other hand, some techniques used in digital systems make those systems more vulnerable to single-bit errors. These techniques are acceptable when the underlying bits are reliable enough that such errors are highly unlikely. A single-bit error in audio data stored directly as linear pulse code modulation (such as on a CD-ROM) causes, at worst, a single click. Instead, many people use audio compression to save storage space and download time, even though a single-bit error may corrupt the entire song.

Analog issues in digital circuits

Digital circuits are made from analog components. The design must assure that the analog nature of the components doesn't dominate the desired digital behavior. Digital systems must manage noise and timing margins, parasitic inductances and capacitances, and filter power connections.

Bad designs have intermittent problems such as "glitches", vanishingly-fast pulses that may trigger some logic but not others, "runt pulses" that do not reach valid "threshold" voltages, or unexpected ("undecoded") combinations of logic states.

Additionally, where clocked digital systems interface to analogue systems or systems that are driven from a different clock, the digital system can be subject to metastability where a change to the input violates the set-up time for a digital input latch. This situation will self-resolve, but will take a random time, and while it persists can result in invalid signals being propagated within the digital system for a short time.

Since digital circuits are made from analog components, digital circuits calculate more slowly than low-precision analog circuits that use a similar amount of space and power. However, the digital circuit will calculate more repeatably, because of its high noise immunity. On the other hand, in the high-precision domain (for example, where 14 or more bits of precision are needed), analog circuits require much more power and area than digital equivalents.

Construction

A digital circuit is often constructed from small electronic circuits called logic gates that can be used to create combinational logic. Each logic gate represents a function of boolean logic. A logic gate is an arrangement of electrically controlled switches, better known as transistors.

Each logic symbol is represented by a different shape. The actual set of shapes was introduced in 1984 under IEEE\ANSI standard 91-1984. "The logic symbol given under this standard are being increasingly used now and have even started appearing in the literature published by manufacturers of digital integrated circuits."

The output of a logic gate is an electrical flow or voltage, that can, in turn, control more logic gates.

Logic gates often use the fewest number of transistors in order to reduce their size, power consumption and cost, and increase their reliability.

Integrated circuits are the least expensive way to make logic gates in large volumes. Integrated circuits are usually designed by engineers using electronic design automation software.

Another form of digital circuit is constructed from lookup tables, (many sold as "programmable logic devices", though other kinds of PLDs exist). Lookup tables can perform the same functions as machines based on logic gates, but can be easily reprogrammed without changing the wiring. This means that a designer can often repair design errors without changing the arrangement of wires. Therefore, in small volume products, programmable logic devices are often the preferred solution. They are usually designed by engineers using electronic design automation software.

When the volumes are medium to large, and the logic can be slow, or involves complex algorithms or sequences, often a small microcontroller is programmed to make an embedded system. These are usually programmed by software engineers.

When only one digital circuit is needed, and its design is totally customized, as for a factory production line controller, the conventional solution is a programmable logic controller, or PLC. These are usually programmed by electricians, using ladder logic.

Structure of digital systems

Engineers use many methods to minimize logic functions, in order to reduce the circuit's complexity. When the complexity is less, the circuit also has fewer errors and less electronics, and is therefore less expensive.

The most widely used simplification is a minimization algorithm like the Espresso heuristic logic minimizer within a CAD system, although historically, binary decision diagrams, an automated Quine–McCluskey algorithm, truth tables, Karnaugh Maps, and Boolean algebra have been used.

Representations are crucial to an engineer's design of digital circuits. Some analysis methods only work with particular representations.

The classical way to represent a digital circuit is with an equivalent set of logic gates. Another way, often with the least electronics, is to construct an equivalent system of electronic switches (usually transistors). One of the easiest ways is to simply have a memory containing a truth table. The inputs are fed into the address of the memory, and the data outputs of the memory become the outputs.

For automated analysis, these representations have digital file formats that can be processed by computer programs. Most digital engineers are very careful to select computer programs ("tools") with compatible file formats.

To choose representations, engineers consider types of digital systems. Most digital systems divide into "combinational systems" and "sequential systems." A combinational system always presents the same output when given the same inputs. It is basically a representation of a set of logic functions, as already discussed.

A sequential system is a combinational system with some of the outputs fed back as inputs. This makes the digital machine perform a "sequence" of operations. The simplest sequential system is probably a flip flop, a mechanism that represents a binary digit or "bit".

Sequential systems are often designed as state machines. In this way, engineers can design a system's gross behavior, and even test it in a simulation, without considering all the details of the logic functions.

Sequential systems divide into two further subcategories. "Synchronous" sequential systems change state all at once, when a "clock" signal changes state. "Asynchronous" sequential systems propagate changes whenever inputs change. Synchronous sequential systems are made of well-characterized asynchronous circuits such as flip-flops, that change only when the clock changes, and which have carefully designed timing margins.

The usual way to implement a synchronous sequential state machine is to divide it into a piece of combinational logic and a set of flip flops called a "state register." Each time a clock signal ticks, the state register captures the feedback generated from the previous state of the combinational logic, and feeds it back as an unchanging input to the combinational part of the state machine. The fastest rate of the clock is set by the most time-consuming logic calculation in the combinational logic.

The state register is just a representation of a binary number. If the states in the state machine are numbered (easy to arrange), the logic function is some combinational logic that produces the number of the next state.

In comparison, asynchronous systems are very hard to design because all possible states, in all possible timings must be considered. The usual method is to construct a table of the minimum and maximum time that each such state can exist, and then adjust the circuit to minimize the number of such states, and force the circuit to periodically wait for all of its parts to enter a compatible state (this is called "self-resynchronization"). Without such careful design, it is easy to accidentally produce asynchronous logic that is "unstable", that is, real electronics will have unpredictable results because of the cumulative delays caused by small variations in the values of the electronic components. Certain circuits (such as the synchronizer flip-flops, switch debouncers, arbiters, and the like which allow external unsynchronized signals to enter synchronous logic circuits) are inherently asynchronous in their design and must be analyzed as such.

As of 2005, almost all digital machines are synchronous designs because it is much easier to create and verify a synchronous design—the software currently used to simulate digital machines does not yet handle asynchronous designs. However, asynchronous logic is thought to be superior, if it can be made to work, because its speed is not constrained by an arbitrary clock; instead, it runs at the maximum speed of its logic gates. Building an asynchronous circuit using faster parts makes the circuit faster.

Many digital systems are data flow machines. These are usually designed using synchronous register transfer logic, using hardware description languages such as VHDL or Verilog.

In register transfer logic, binary numbers are stored in groups of flip flops called registers. The outputs of each register are a bundle of wires called a "bus" that carries that number to other calculations. A calculation is simply a piece of combinational logic. Each calculation also has an output bus, and these may be connected to the inputs of several registers. Sometimes a register will have a multiplexer on its input, so that it can store a number from any one of several buses. Alternatively, the outputs of several items may be connected to a bus through buffers that can turn off the output of all of the devices except one. A sequential state machine controls when each register accepts new data from its input.

In the 1980s, some researchers discovered that almost all synchronous register-transfer machines could be converted to asynchronous designs by using first-in-first-out synchronization logic. In this scheme, the digital machine is characterized as a set of data flows. In each step of the flow, an asynchronous "synchronization circuit" determines when the outputs of that step are valid, and presents a signal that says, "grab the data" to the stages that use that stage's inputs. It turns out that just a few relatively simple synchronization circuits are needed.

The most general-purpose register-transfer logic machine is a computer. This is basically an automatic binary abacus. The control unit of a computer is usually designed as a microprogram run by a microsequencer. A microprogram is much like a player-piano roll. Each table entry or "word" of the microprogram commands the state of every bit that controls the computer. The sequencer then counts, and the count addresses the memory or combinational logic machine that contains the microprogram. The bits from the microprogram control the arithmetic logic unit, memory and other parts of the computer, including the microsequencer itself.

In this way, the complex task of designing the controls of a computer is reduced to a simpler task of programming a collection of much simpler logic machines.

Computer architecture is a specialized engineering activity that tries to arrange the registers, calculation logic, buses and other parts of the computer in the best way for some purpose. Computer architects have applied large amounts of ingenuity to computer design to reduce the cost and increase the speed and immunity to programming errors of computers. An increasingly common goal is to reduce the power used in a battery-powered computer system, such as a cell-phone. Many computer architects serve an extended apprenticeship as microprogrammers.

"Specialized computers" are usually a conventional computer with a special-purpose microprogram.

Automated design tools

To save costly engineering effort, much of the effort of designing large logic machines has been automated. The computer programs are called "electronic design automation tools" or just "EDA."

Simple truth table-style descriptions of logic are often optimized with EDA that automatically produces reduced systems of logic gates or smaller lookup tables that still produce the desired outputs. The most common example of this kind of software is the Espresso heuristic logic minimizer.

Most practical algorithms for optimizing large logic systems use algebraic manipulations or binary decision diagrams, and there are promising experiments with genetic algorithms and annealing optimizations.

To automate costly engineering processes, some EDA can take state tables that describe state machines and automatically produce a truth table or a function table for the combinational logic of a state machine. The state table is a piece of text that lists each state, together with the conditions controlling the transitions between them and the belonging output signals.

It is common for the function tables of such computer-generated state-machines to be optimized with logic-minimization software such as Minilog.

Often, real logic systems are designed as a series of sub-projects, which are combined using a "tool flow." The tool flow is usually a "script," a simplified computer language that can invoke the software design tools in the right order.

Tool flows for large logic systems such as microprocessors can be thousands of commands long, and combine the work of hundreds of engineers.

Writing and debugging tool flows is an established engineering specialty in companies that produce digital designs. The tool flow usually terminates in a detailed computer file or set of files that describe how to physically construct the logic. Often it consists of instructions to draw the transistors and wires on an integrated circuit or a printed circuit board.

Parts of tool flows are "debugged" by verifying the outputs of simulated logic against expected inputs. The test tools take computer files with sets of inputs and outputs, and highlight discrepancies between the simulated behavior and the expected behavior.

Once the input data is believed correct, the design itself must still be verified for correctness. Some tool flows verify designs by first producing a design, and then scanning the design to produce compatible input data for the tool flow. If the scanned data matches the input data, then the tool flow has probably not introduced errors.

The functional verification data are usually called "test vectors." The functional test vectors may be preserved and used in the factory to test that newly constructed logic works correctly. However, functional test patterns don't discover common fabrication faults. Production tests are often designed by software tools called "test pattern generators". These generate test vectors by examining the structure of the logic and systematically generating tests for particular faults. This way the fault coverage can closely approach 100%, provided the design is properly made testable.

Once a design exists, and is verified and testable, it often needs to be processed to be manufacturable as well. Modern integrated circuits have features smaller than the wavelength of the light used to expose the photoresist. Manufacturability software adds interference patterns to the exposure masks to eliminate open-circuits, and enhance the masks' resolution and contrast.

Design for testability

"There are several reasons for testing a logic circuit. When the circuit is first developed, it is necessary to verify that the design circuit meets the required functional and timing specifications. When multiple copies of a correctly designed circuit are being manufactured, it is essential to test each copy to ensure that the manufacturing process has not introduced any flaws.

A large logic machine (say, with more than a hundred logical variables) can have an astronomical number of possible states. Obviously, in the factory, testing every state is impractical if testing each state takes a microsecond, and there are more states than the number of microseconds since the universe began. Unfortunately, this ridiculous-sounding case is typical.

Fortunately, large logic machines are almost always designed as assemblies of smaller logic machines. To save time, the smaller sub-machines are isolated by permanently-installed "design for test" circuitry, and are tested independently.

One common test scheme known as "scan design" moves test bits serially (one after another) from external test equipment through one or more serial shift registers known as "scan chains". Serial scans have only one or two wires to carry the data, and minimize the physical size and expense of the infrequently-used test logic.

After all the test data bits are in place, the design is reconfigured to be in "normal mode" and one or more clock pulses are applied, to test for faults (e.g. stuck-at low or stuck-at high) and capture the test result into flip-flops and/or latches in the scan shift register(s).

Finally, the result of the test is shifted out to the block boundary and compared against the predicted "good machine" result.

In a board-test environment, serial to parallel testing has been formalized with a standard called "JTAG" (named after the "Joint Test Action Group" that proposed it).

Another common testing scheme provides a test mode that forces some part of the logic machine to enter a "test cycle." The test cycle usually exercises large independent parts of the machine.

Trade-offs

Several numbers determine the practicality of a system of digital logic. Engineers explored numerous electronic devices to get an ideal combination of fanout, speed, low cost and reliability.

The cost of a logic gate is crucial. In the 1930s, the earliest digital logic systems were constructed from telephone relays because these were inexpensive and relatively reliable. After that, engineers always used the cheapest available electronic switches that could still fulfill the requirements.

The earliest integrated circuits were a happy accident. They were constructed not to save money, but to save weight, and permit the Apollo Guidance Computer to control an inertial guidance system for a spacecraft. The first integrated circuit logic gates cost nearly $50 (in 1960 dollars, when an engineer earned $10,000/year). To everyone's surprise, by the time the circuits were mass-produced, they had become the least-expensive method of constructing digital logic. Improvements in this technology have driven all subsequent improvements in cost.

With the rise of integrated circuits, reducing the absolute number of chips used represented another way to save costs. The goal of a designer is not just to make the simplest circuit, but to keep the component count down. Sometimes this results in slightly more complicated designs with respect to the underlying digital logic but nevertheless reduces the number of components, board size, and even power consumption.

For example, in some logic families, NAND gates are the simplest digital gate to build. All other logical operations can be implemented by NAND gates. If a circuit already required a single NAND gate, and a single chip normally carried four NAND gates, then the remaining gates could be used to implement other logical operations like logical and. This could eliminate the need for a separate chip containing those different types of gates.

The "reliability" of a logic gate describes its mean time between failure (MTBF). Digital machines often have millions of logic gates. Also, most digital machines are "optimized" to reduce their cost. The result is that often, the failure of a single logic gate will cause a digital machine to stop working.

Digital machines first became useful when the MTBF for a switch got above a few hundred hours. Even so, many of these machines had complex, well-rehearsed repair procedures, and would be nonfunctional for hours because a tube burned-out, or a moth got stuck in a relay. Modern transistorized integrated circuit logic gates have MTBFs greater than 82 billion hours (8.2×10^{10}) hours, and need them because they have so many logic gates.

Fanout describes how many logic inputs can be controlled by a single logic output without exceeding the current ratings of the gate. The minimum practical fanout is about five. Modern electronic logic using CMOS transistors for switches have fanouts near fifty, and can sometimes go much higher.

The "switching speed" describes how many times per second an inverter (an electronic representation of a "logical not" function) can change from true to false and back. Faster logic can accomplish more operations in less time. Digital logic first became useful when switching speeds got above fifty hertz, because that was faster than a team of humans operating mechanical calculators. Modern electronic digital logic routinely switches at five gigahertz (5×10^9 hertz), and some laboratory systems switch at more than a terahertz (1×10^{12} hertz).

Logic families

Design started with relays. Relay logic was relatively inexpensive and reliable, but slow. Occasionally a mechanical failure would occur. Fanouts were typically about ten, limited by the resistance of the coils and arcing on the contacts from high voltages.

Later, vacuum tubes were used. These were very fast, but generated heat, and were unreliable because the filaments would burn out. Fanouts were typically five to seven, limited by the heating from the tubes' current. In the 1950s, special "computer tubes" were developed with filaments that omitted volatile elements like silicon. These ran for hundreds of thousands of hours.

The first semiconductor logic family was resistor-transistor logic. This was a thousand times more reliable than tubes, ran cooler, and used less power, but had a very low fan-in of three. Diode-transistor logic improved the fanout up to about seven, and reduced the power. Some DTL designs used two power-supplies with alternating layers of NPN and PNP transistors to increase the fanout.

Transistor transistor logic (TTL) was a great improvement over these. In early devices, fanout improved to ten, and later variations reliably achieved twenty. TTL was also fast, with some variations achieving switching times as low as twenty nanoseconds. TTL is still used in some designs.

Emitter coupled logic is very fast but uses a lot of power. It was extensively used for high-performance computers made up of many medium-scale components (such as the Illiac IV).

By far, the most common digital integrated circuits built today use CMOS logic, which is fast, offers high circuit density and low-power per gate.

Non-electronic logic

It is possible to construct non-electronic digital mechanisms. In principle, any technology capable of representing discrete states and representing logic operations could be used to build mechanical logic. MIT students Erlyne Gee, Edward Hardebeck, Danny Hillis (co-author of The Connection Machine), Margaret Minsky and brothers Barry and Brian Silverman, built two working computers from Tinker toys, string, a brick, and a sharpened pencil. This "Tinkertoy computer" is exhibited in the Boston Museum of Science.

Hydraulic, pneumatic and mechanical versions of logic gates exist and are used in situations where electricity cannot be used. The first two types are considered under the heading of fluidics. One application of fluidic logic is in military hardware that is likely to be exposed to a nuclear electromagnetic pulse (nuclear EMP, or NEMP) that would destroy electrical circuits.

Mechanical logic is frequently used in inexpensive controllers, such as those in washing machines. The first computer design, by Charles Babbage, was designed to use mechanical logic. Mechanical logic might also be used in very small computers that could be built by nanotechnology.

Another example is that if two particular enzymes are required to prevent the construction of a particular protein, this is the equivalent of a biological "NAND" gate.

Recent developments

The discovery of superconductivity has enabled the development of Rapid Single Flux Quantum (RSFQ) circuit technology, which uses Josephson junctions instead of transistors. Most recently, attempts are being made to construct purely optical computing systems capable of processing digital information using nonlinear optical elements.

Chapter 5

Expert System

An **expert system** is software that uses a knowledge base of human expertise for problem solving, or clarify uncertainties where normally one or more human experts would need to be consulted. Expert systems are most common in a specific problem domain, and are a traditional application and/or subfield of artificial intelligence (AI). A wide variety of methods can be used to simulate the performance of the expert; however, common to most or all are: 1) the creation of a knowledge base which uses some knowledge representation structure to capture the knowledge of the Subject Matter Expert (SME); 2) a process of gathering that knowledge from the SME and codifying it according to the structure, which is called knowledge engineering; and 3) once the system is developed, it is placed in the same real world problem solving situation as the human SME, typically as an aid to human workers or as a supplement to some information system. Expert systems may or may not have learning components.

Expert systems were introduced by researchers in the Stanford Heuristic Programming Project, including the "father of expert systems" Edward Feigenbaum, with the Dendral and Mycin systems. Principal contributors to the technology were Bruce Buchanan, Edward Shortliffe, Randall Davis, William vanMelle, Carli Scott, and others at Stanford. Expert systems were among the first truly successful forms of AI software.

The topic of expert systems also has connections to general systems theory, operations research, business process reengineering, and various topics in applied mathematics and management science.

Structure

Software architecture

The following general points about expert systems and their architecture have been outlined:

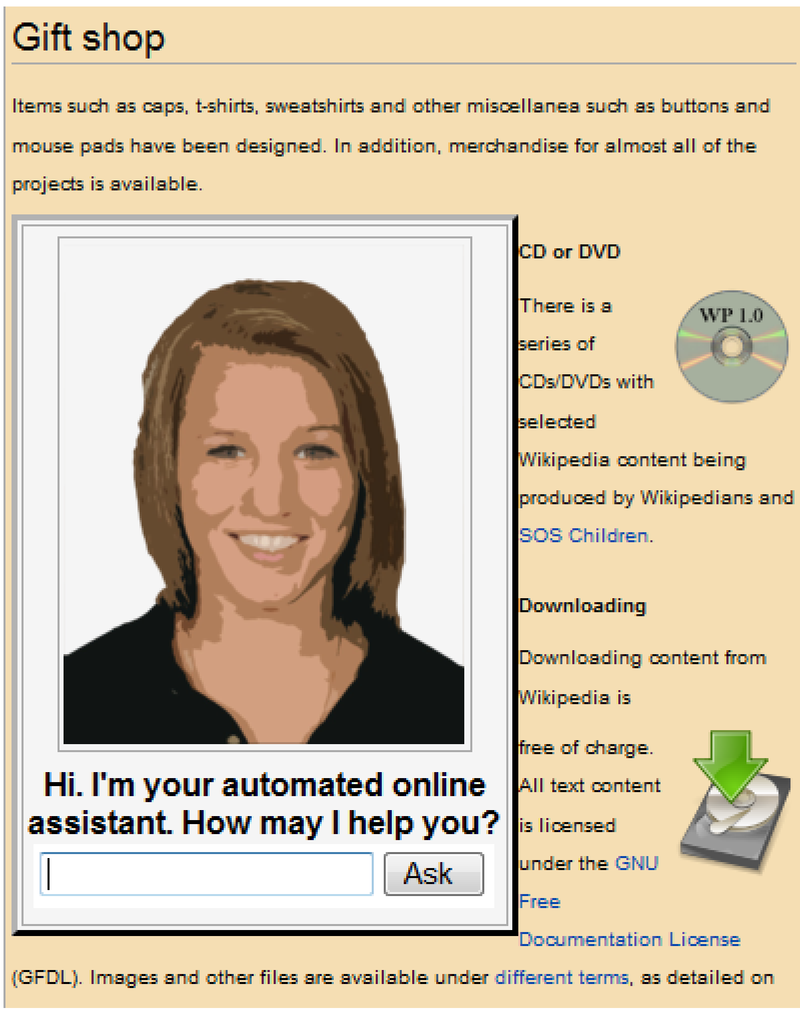
1. The sequence of steps taken to reach a conclusion is dynamically synthesized with each new case. The sequence is not explicitly programmed at the time that the system is built.
2. Expert systems can process multiple values for any problem parameter. This permits more than one line of reasoning to be pursued and the results of incomplete (not fully determined) reasoning to be presented.
3. Problem solving is accomplished by applying specific knowledge rather than specific technique. This is a key idea in expert systems technology. It reflects the belief that human experts do not process their knowledge differently from others, but they do possess different knowledge. With this philosophy, when one finds that their expert system does not produce the desired results, work begins to expand the knowledge base, not to re-program the procedures.

There are various expert systems in which a rulebase and an inference engine cooperate to simulate the reasoning process that a human expert pursues in analyzing a problem and arriving at a conclusion. In these systems, in order to simulate the human reasoning process, a vast amount of knowledge needs to be stored in the knowledge base. Generally, the knowledge base of such an expert system consists of a relatively large number of "if/then" type statements that are interrelated in a manner that, in theory at least, resembles the sequence of mental steps that are involved in the human reasoning process.

Because of the need for large storage capacities and related programs to store the rulebase, most expert systems have, in the past, been run only on large information handling systems. Recently, the storage capacity of personal computers has increased to a point to which it is becoming possible to consider running some types of simple expert systems on personal computers.

In some applications of expert systems, the nature of the application and the amount of stored information necessary to simulate the human reasoning process for that application is too vast to store in the active memory of a computer. In other applications of expert systems, the nature of the application is such that not all of the information is always needed in the reasoning process. An example of this latter type of application would be the use of an expert system to diagnose a data processing system comprising many separate components, some of which are optional. When that type of expert system employs a single integrated rulebase to diagnose the minimum system configuration of the data processing system, much of the rulebase is not required since many of the optional components will not be present in the system. Nevertheless, early expert systems required the entire rulebase to be stored since all the rules were, in effect, chained or linked together by the structure of the rulebase.

When the rulebase is segmented, preferably into contextual segments or units, it is then possible to eliminate the portions of the rulebase containing data or knowledge that is not needed in a particular application. The segmentation of the rulebase also allows the expert system to be run on or with systems having much smaller memory capacities than was possible with earlier arrangements, since each segment of the rulebase can be paged into and out of the system as needed. Segmentation into contextual units requires that the expert system manage various intersegment relationships as segments are paged into and out of memory during the execution of the program. Since the system permits a rulebase segment to be called and executed at any time during the processing of the first rulebase, provisions must be made to store the data that has accumulated up to that point so that later in the process, when the system returns to the first segment, it can proceed from the last point or rule node that was processed. Also, provisions must be made so that data that has been collected by the system up to that point can be passed onto the second segment of the rulebase after it has been paged into the system, and data collected during the processing of the second segment can be passed to the first segment when the system returns to complete processing that segment.



Avatars and dialog systems can enhance the user interface of expert systems, as for this automated online assistant.

The user interface is important in the information collection process.

Comparison to problem-solving systems

The principal distinction between expert systems and traditional problem solving programs is the way in which the problem related expertise is coded. In traditional applications, problem-related expertise is encoded in both program and data structures. In the expert system approach all of the problem expertise is encoded *mostly* in data structures.

In an example related to tax advice, the traditional approach has data structures that describe the taxpayer and tax tables, and a program that contains rules (encoding expert knowledge) that relate information about the taxpayer to tax table choices. In contrast, in the expert system approach, the latter information is also encoded in data structures. The collective data structures are called the knowledge base. The program (inference engine) of an expert system is relatively independent of the problem domain (taxes) and processes the rules without regard to the problem area they describe.

This organization has several benefits:

- New rules can be added to the knowledge base or altered without needing to rebuild the program. This allows changes to be made rapidly to a system (e.g., after it has been shipped to its customers, to accommodate very recent changes in state or federal tax codes).
- Rules are arguably easier for (non-programmer) domain experts to create and modify than writing code. Commercial rule engines typically come with editors that allow rule creation/modification through a graphical user interface, which also performs actions such as consistency and redundancy checks.

Modern rule engines allow a hybrid approach: some allow rules to be "compiled" into a form that is more efficiently machine-executable. Also, for efficiency concerns, rule engines allow rules to be defined more expressively and concisely by allowing software developers to create functions in a traditional programming language such as Java, which can then be invoked from either the condition or the action of a rule. Such functions may incorporate domain-specific (but reusable) logic.

Methods of operation

Certainty factors

One method of operation of expert systems is through a quasi-probabilistic approach with *certainty factors*:

A human, when reasoning, does not always make statements with 100% confidence: he might venture, "If Fritz is green, then he is probably a frog" (after all, he might be a chameleon). This type of reasoning can be imitated using numeric values called *confidences*. For example, if it is known that Fritz is green, it might be concluded with 0.85 confidence that he is a frog; or, if it is known that he is a frog, it might be concluded with 0.95 confidence that he hops. These certainty factor (CF) numbers quantify uncertainty in the degree to which the available evidence supports a hypothesis. They represent a degree of confirmation, and are not probabilities in a Bayesian sense. The CF calculus, developed by Shortliffe & Buchanan, increases or decreases the CF associated with a hypothesis as each new piece of evidence becomes available. It can be mapped to a probability update, although degrees of confirmation are not expected to obey the laws of probability. It is important to note, for example, that evidence for hypothesis H may have nothing to contribute to the degree to which Not_h is confirmed or disconfirmed (e.g., although a fever lends some support to a diagnosis of infection, fever does not disconfirm alternative hypotheses) and that the sum of CFs of many competing hypotheses may be greater than one (i.e., many hypotheses may be well confirmed based on available evidence).

The CF approach to a rule-based expert system design does not have a widespread following, in part because of the difficulty of meaningfully assigning CFs a priori. (The above example of green creatures being likely to be frogs is excessively naive.) Alternative approaches to quasi-probabilistic reasoning in expert systems involve fuzzy logic, which has a firmer mathematical foundation. Also, rule-engine shells such as Drools and Jess do not support probability manipulation: they use an alternative mechanism called salience, which is used to prioritize the order of evaluation of activated rules.

In certain areas, as in the tax-advice scenarios discussed below, probabilistic approaches are not acceptable. For instance, a 95% probability of being correct means a 5% probability of being wrong. The rules that are defined in such systems have no exceptions: they are only a means of achieving software flexibility when external circumstances change frequently. Because rules are stored as data, the core software does not need to be rebuilt each time changes to federal and state tax codes are announced.

Certainty factors were introduced by the MYCIN rule-based expert system.

Chaining

Two methods of reasoning when using inference rules are forward chaining and backward chaining.

Forward chaining starts with the data available and uses the inference rules to extract more data until a desired goal is reached. An inference engine using forward chaining searches the inference rules until it finds one in which the if clause is known to be true. It then concludes the *then* clause and adds this information to its data. It continues to do this until a goal is reached. Because the data available determines which inference rules are used, this method is also classified as *data driven*.

Backward chaining starts with a list of goals and works backwards to see if there is data which will allow it to conclude any of these goals. An inference engine using backward chaining would search the inference rules until it finds one which has a *then* clause that matches a desired goal. If the *if* clause of that inference rule is not known to be true, then it is added to the list of goals. For example, suppose a rule base contains

1. (1) IF X is green THEN X is a frog. (Confidence Factor: +1%)
2. (2) IF X is NOT green THEN X is NOT a frog. (Confidence Factor: +99%)
3. (3) IF X is a frog THEN X hops. (Confidence Factor: +50%)
4. (4) IF X is NOT a frog THEN X does NOT hop. (Confidence Factor +50%)

Suppose a goal is to conclude that Fritz hops. Let X = "Fritz". The rule base would be searched and rule (3) would be selected because its conclusion (the *then* clause) matches the goal. It is not known that Fritz is a frog, so this "if" statement is added to the goal list. The rule base is again searched and this time rule (1) is selected because its then clause matches the new goal just added to the list. This time, the *if* clause (Fritz is green) is known to be true and the goal that Fritz hops is concluded. Because the list of goals determines which rules are selected and used, this method is called *goal driven*.

However, note that if we use confidence factors in even a simplistic fashion - for example, by multiplying them together as if they were like soft probabilities - we get a result that is known with a confidence factor of only one-half of 1%. (This is by multiplying 0.5 x 0.01 = 0.005). This is useful, because without confidence factors, we might erroneously conclude with certainty that a sea turtle named Fritz hops just by virtue of being green. In Classical logic or Aristotelian term logic systems, there are no probabilities or confidence factors; all facts are regarded as certain. An ancient example from Aristotle states, "Socrates is a man. All men are mortal. Thus Socrates is mortal."

In real world applications, few facts are known with absolute certainty and the opposite of a given statement may be more likely to be true ("Green things in the pet store are not frogs, with the probability or confidence factor of 99% in my pet store survey"). Thus it is often useful when building such systems to try and prove both the goal and the opposite of a given goal to see which is more likely.

Inference rules

An inference rule is a conditional statement with two parts: an *if* clause and a *then* clause. This rule is what gives expert systems the ability to find solutions to diagnostic and prescriptive problems. An example of an inference rule is:

If the restaurant choice includes French and the occasion is romantic,
Then the restaurant choice is definitely Paul Bocuse.

An expert system's rulebase is made up of many such inference rules. They are entered as separate rules and it is the inference engine that uses them together to draw conclusions. Because each rule is a unit, rules may be deleted or added without affecting other rules - though it should affect which conclusions are reached. One advantage of inference rules over traditional programming is that inference rules use reasoning which more closely resembles human reasoning.

Thus, when a conclusion is drawn, it is possible to understand how this conclusion was reached. Furthermore, because the expert system uses knowledge in a form similar to the that of the expert, it may be easier to retrieve this information directly from the expert.

Real-time adaption

Industrial processes, data networks, and many other systems change their state and even their structure over time. Real time expert systems are designed to reason over time and change conclusions as the monitored system changes. Most of these systems must respond to constantly changing input data, arriving automatically from other systems such as process control systems or network management systems.

Representation includes features for defining changes in belief of data or conclusions over time. This is necessary because data becomes stale. Approaches to this can include decaying belief functions, or the simpler validity interval that simply lets data and conclusions expire after specified time period, falling to "unknown" until refreshed. An often-cited example (attributed to real time expert system pioneer Robert L. Moore) is a hypothetical expert system that might be used to drive a car. Based on video input, there might be an intermediate conclusion that a stop light is green and a final conclusion that it is OK to drive through the intersection. But that data and the subsequent conclusions have a very limited lifetime. You would not want to be a passenger in a car driven based on data and conclusions that were, say, an hour old.

The inference engine must track the times of each data input and each conclusion, and propagate new information as it arrives. It must ensure that all conclusions are still current. Facilities for periodically scanning data, acquiring data on demand, and filtering noise, become essential parts of the overall system. Facilities to reason within a fixed deadline are important in many of these applications.

An overview of requirements for a real-time expert system shell is given in. Examples of real time expert system applications are given in and. Several conferences were dedicated to real time expert system applications in the chemical process industries, including.

Ability to make relevant inquiries

An additional skill of an expert system is the ability to give relevant inquiries based on previous input from a human user, in order to give better replies or other actions, as well as working faster, which also pleases an impatient or busy human user - it allows a priori volunteering of information that the user considers important.

Also, the user may choose not to respond to every question, forcing the expert system to function in the presence of partial information.

Commercially viable systems will try to optimize the user experience by presenting options for commonly requested information based on a history of previous queries of the system using technology such as forms, augmented by keyword-based search. The gathered information may be verified by a confirmation step (e.g., to recover from spelling mistakes), and now act as an input into a forward-chaining engine. If confirmatory questions are asked in a subsequent phase, based on the rules activated by the obtained information, they are more likely to be specific and relevant. Such abilities can largely be achieved by control flow structures.

In an expert system, implementing the ability to learn from a stored history of its previous use involves employing technologies considerably different from that of rule engines, and is considerably more challenging from a software-engineering perspective. It can, however, make the difference between commercial success and failure. A large part of the revulsion that users felt towards Microsoft's Office Assistant was due to the extreme naivete of its rules ("It looks like you are typing a letter: would you like help?") and its failure to adapt to the user's level of expertise over time (e.g. a user who regularly uses features such as Styles, Outline view, Table of Contents or cross-references is unlikely to be a beginner who needs help writing a letter).

Explanation system

Another major distinction between expert systems and traditional systems is illustrated by the following answer given by the system when the user answers a question with another question, "Why", as occurred in the above example. The answer is:

A. I am trying to determine the type of restaurant to suggest. So far Indian is not a likely choice. It is possible that French is a likely choice. If I know that if the diner is a wine drinker, and the preferred wine is French, then there is strong evidence that the restaurant choice should include French.

It is very difficult to implement a general explanation system (answering questions like "Why" and "How") in a traditional computer program. An expert system can generate an explanation by retracing the steps of its reasoning. The response of the expert system to the question "Why" exposes the underlying knowledge structure. It is a rule; a set of antecedent conditions which, if true, allow the assertion of a consequent. The rule references values, and tests them against various constraints or asserts constraints onto them. This, in fact, is a significant part of the knowledge structure. There are values, which may be associated with some organizing entity. For example, the individual diner is an entity with various attributes (values) including whether they drink wine and the kind of wine. There are also rules, which associate the currently known values of some attributes with assertions that can be made about other attributes. It is the orderly processing of these rules that dictates the dialogue itself.

Shells

A shell (or inference engine) is a complete development environment for building and maintaining knowledge-based applications. It provides a step-by-step methodology, and ideally a user-friendly interface such as a graphical interface, for a knowledge engineer that allows the domain experts themselves to be directly involved in structuring and encoding the knowledge. Examples of shells include Drools, CLIPS, JESS, d3web, G2, eGanges, and OpenKBM (initially developed as a replacement for G2).

Knowledge engineering

The building, maintaining and development of expert systems is known as *knowledge engineering*. Knowledge engineering is a "discipline that involves integrating knowledge into computer systems in order to solve complex problems normally requiring a high level of human expertise".

There are generally three individuals having an interaction in an expert system. Primary among these is the end-user, the individual who uses the system for its problem solving assistance. In the construction and maintenance of the system there are two other roles: the problem domain expert who builds the system and supplies the knowledge base, and a knowledge engineer who assists the experts in determining the representation of their knowledge, enters this knowledge into an explanation module and who defines the inference technique required to solve the problem. Usually the knowledge engineer will represent the problem solving activity in the form of rules. When these rules are created from domain expertise, the knowledge base stores the rules of the expert system.

Applications

General types of problems solved

Expert systems are most valuable to organizations that have a high-level of know-how experience and expertise that cannot be easily transferred to other members. They are designed to carry the intelligence and information found in the intellect of experts and provide this knowledge to other members of the organization for problem-solving purposes.

Typically, the problems to be solved are of the sort that would normally be tackled by a professional, such as a medical professional in the case of clinical decision support systems. Real experts in the problem domain (which will typically be very narrow, for instance "diagnosing skin conditions in teenagers") are asked to provide "rules of thumb" on how they evaluate the problem — either explicitly with the aid of experienced systems developers, or sometimes implicitly, by getting such experts to evaluate test cases and using computer programs to examine the test data and derive rules from that (in a strictly limited manner). Generally, expert systems are used for problems for which there is no single "correct" solution which can be encoded in a conventional algorithm — one would not write an expert system to find the shortest paths through graphs, or to sort data, as there are simpler ways to do these tasks.

Simple systems use simple true/false logic to evaluate data. More sophisticated systems are capable of performing at least some evaluation, taking into account real-world uncertainties, using such methods as fuzzy logic. Such sophistication is difficult to develop and still highly imperfect.

Examples of applications

Expert systems are designed to facilitate tasks in the fields of accounting, medicine, process control, financial service, production, human resources, among others. Typically, the problem area is complex enough that a more simple traditional algorithm cannot provide a proper solution. The foundation of a successful expert system depends on a series of technical procedures and development that may be designed by technicians and related experts. As such, expert systems do not typically provide a definitive answer, but provide probabilistic recommendations.

An example of the application of expert systems in the financial field is expert systems for mortgages. Loan departments are interested in expert systems for mortgages because of the growing cost of labour, which makes the handling and acceptance of relatively small loans less profitable. They also see a possibility for standardised, efficient handling of mortgage loan by applying expert systems, appreciating that for the acceptance of mortgages there are hard and fast rules which do not always exist with other types of loans. Another common application in the financial area for expert systems are in trading recommendations in various marketplaces. These markets involve numerous variables and human emotions which may be impossible to deterministically characterize, thus expert systems based on the rules of thumb from experts and simulation data are used. Expert system of this type can range from ones providing regional retail recommendations, like Wishabi, to ones used to assist monetary decisions by financial institutions and governments.

Another 1970s and 1980s application of expert systems, which we today would simply call AI, was in computer games. For example, the computer baseball games Earl Weaver Baseball and Tony La Russa Baseball each had highly detailed simulations of the game strategies of those two baseball managers. When a human played the game against the computer, the computer queried the Earl Weaver or Tony La Russa Expert System for a decision on what strategy to follow. Even those choices where some randomness was part of the natural system (such as when to throw a surprise pitch-out to try to trick a runner trying to steal a base) were decided based on probabilities supplied by Weaver or La Russa. Today we would simply say that "the game's AI provided the opposing manager's strategy."

Advantages and disadvantages

Advantages

- Compared to traditional programming techniques, expert-system approaches provide the added flexibility (and hence easier modifiability) with the ability to model rules as data rather than as code. In situations where an organization's IT department is overwhelmed by a software-development backlog, rule-engines, by facilitating turnaround, provide a means that can allow organizations to adapt more readily to changing needs.

- In practice, modern expert-system technology is employed as an adjunct to traditional programming techniques, and this hybrid approach allows the combination of the strengths of both approaches. Thus, rule engines allow control through programs (and user interfaces) written in a traditional language, and also incorporate necessary functionality such as inter-operability with existing database technology.

Disadvantages

- The Garbage In, Garbage Out (GIGO) phenomenon: A system that uses expert-system technology provides no guarantee about the quality of the rules on which it operates. All self-designated "experts" are not necessarily so, and one notable challenge in expert system design is in getting a system to recognize the limits to its knowledge.
- Expert systems are notoriously narrow in their domain of knowledge — as an amusing example, a researcher used the "skin disease" expert system to diagnose his rustbucket car as likely to have developed measles — and the systems are thus prone to making errors that humans would easily spot. Additionally, once some of the mystique had worn off, most programmers realized that simple expert systems were essentially just slightly more elaborate versions of the decision logic they had already been using. Therefore, some of the techniques of expert systems can now be found in most complex programs without drawing much recognition.
- An expert system or rule-based approach is not optimal for all problems, and considerable knowledge is required so as to not misapply the systems.
- Ease of rule creation and rule modification can be double-edged. A system can be sabotaged by a non-knowledgeable user who can easily add worthless rules or rules that conflict with existing ones. Reasons for the failure of many systems include the absence of (or neglect to employ diligently) facilities for system audit, detection of possible conflict, and rule lifecycle management (e.g. version control, or thorough testing before deployment). The problems to be addressed here are as much technological as organizational.

An example and a good demonstration of the limitations of an expert system is the Windows operating system troubleshooting software located in the "help" section in the taskbar menu. Obtaining technical operating system support is often difficult for individuals not closely involved with the development of the operating system. Microsoft has designed their expert system to provide solutions, advice, and suggestions to common errors encountered while using their operating systems.

Chapter 6

Digital Library

A **digital library** is a library in which collections are stored in digital formats (as opposed to print, microform, or other media) and accessible by computers. The digital content may be stored locally, or accessed remotely via computer networks. A digital library is a type of information retrieval system.

The *DELOS Digital Library Reference Model* defines a digital library as:

An organization, which might be virtual, that comprehensively collects, manages and preserves for the long term rich digital content, and offers to its user communities specialized functionality on that content, of measurable quality and according to codified policies.

The first use of the term *digital library* in print may have been in a 1988 report to the Corporation for National Research Initiatives The term *digital libraries* was first popularized by the NSF/DARPA/NASA Digital Libraries Initiative in 1994. These draw heavily on As We May Think by Vannevar Bush in 1945, which set out a vision not in terms of technology, but user experience. The term *virtual library* was initially used interchangeably with *digital library,* but is now primarily used for libraries that are virtual in other senses (such as libraries which aggregate distributed content).

A distinction is often made between content that was created in a digital format, known as born-digital, and information that has been converted from a physical medium, e.g., paper, by digitizing. The term hybrid library is sometimes used for libraries that have both physical collections and digital collections. For example, American Memory is a digital library within the Library of Congress. Some important digital libraries also serve as long term archives, for example, the Eprint arXiv, and the Internet Archive.

Academic repositories

Many academic libraries are actively involved in building institutional repositories of the institution's books, papers, theses, and other works which can be digitized or were 'born digital'. Many of these repositories are made available to the general public with few restrictions, in accordance with the goals of open access, in contrast to the publication of research in commercial journals, where the publishers often limit access rights. Institutional, truly free, and corporate repositories are sometimes referred to as digital libraries.

Digital archives

Physical archives differ from physical libraries in several ways. Traditionally, archives were defined as:

1. Containing primary sources of information (typically letters and papers directly produced by an individual or organization) rather than the secondary sources found in a library (books, periodicals, etc);
2. Having their contents organized in groups rather than individual items.
3. Having unique contents.

The technology used to create digital libraries has been even more revolutionary for archives since it breaks down the second and third of these general rules. In other words, "digital archives" or "online archives" will still generally contain primary sources, but they are likely to be described individually rather than (or in addition to) in groups or collections, and because they are digital their contents are easily reproducible and may indeed have been reproduced from elsewhere. The Oxford Text Archive is generally considered to be the oldest digital archive of academic physical primary source materials.

The future

Large scale digitization projects are underway at Google, the Million Book Project, and Internet Archive. With continued improvements in book handling and presentation technologies such as optical character recognition and ebooks, and development of alternative depositories and business models, digital libraries are rapidly growing in popularity as demonstrated by Google, Yahoo!, and MSN's efforts. Just as libraries have ventured into audio and video collections, so have digital libraries such as the Internet Archive.

According to Larry Lannom, Director of Information Management Technology at the nonprofit Corporation for National Research Initiatives, “all the problems associated with digital libraries are wrapped up in archiving.” He goes on to state, “If in 100 years people can still read your article, we’ll have solved the problem.” Daniel Akst, author of The Webster Chronicle, proposes that “the future of libraries—and of information—is digital.” Peter Lyman and Hal Varian, information scientists at the University of California, Berkeley, estimate that “the world’s total yearly production of print, film, optical, and magnetic content would require roughly 1.5 billion gigabytes of storage." Therefore, they believe that "soon it will be technologically possible for an average person to access virtually all recorded information."

Searching

Most digital libraries provide a search interface which allows resources to be found. These resources are typically deep web (or invisible web) resources since they frequently cannot be located by search engine crawlers. Some digital libraries create special pages or sitemaps to allow search engines to find all their resources. Digital libraries frequently use the Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH) to expose their metadata to other digital libraries, and search engines like Google Scholar, Yahoo! and Scirus can also use OAI-PMH to find these deep web resources.

There are two general strategies for searching a **federation** of digital libraries:

1. distributed searching, and
2. searching previously harvested metadata.

Distributed searching typically involves a client sending multiple search requests in parallel to a number of servers in the federation. The results are gathered, duplicates are eliminated or clustered, and the remaining items are sorted and presented back to the client. Protocols like Z39.50 are frequently used in distributed searching. A benefit to this approach is that the resource-intensive tasks of indexing and storage are left to the respective servers in the federation. A drawback to this approach is that the search mechanism is limited by the different indexing and ranking capabilities of each database, making it difficult to assemble a combined result consisting of the most relevant found items.

Searching over previously harvested metadata involves searching a locally stored index of information that has previously been collected from the libraries in the federation. When a search is performed, the search mechanism does not need to make connections with the digital libraries it is searching - it already has a local representation of the information. This approach requires the creation of an indexing and harvesting mechanism which operates regularly, connecting to all the digital libraries and querying the whole collection in order to discover new and updated resources. OAI-PMH is frequently used by digital libraries for allowing metadata to be harvested. A benefit to this approach is that the search mechanism has full control over indexing and ranking algorithms, possibly allowing more consistent results. A drawback is that harvesting and indexing systems are more resource-intensive and therefore expensive.

Frameworks

The formal reference models include the DELOS Digital Library Reference Model (Agosti, et al., 2006) and the Streams, Structures, Spaces, Scenarios, Societies (5S) formal framework The Reference Model for an Open Archival Information System (OAIS) provides a framework to address digital preservation.

Construction and organization

Software

There are a number of software packages for use in general digital libraries, for notable ones see Digital library software. Institutional repository software, which focuses primarily on ingest, preservation and access of locally produced documents, particularly locally produced academic outputs, can be found in Institutional repository software.

Digitization

In the past few years, procedures for digitizing books at high speed and comparatively low cost have improved considerably with the result that it is now possible to plan the digitization of millions of books per year for creating digital libraries.

Advantages

The advantages of digital libraries as a means of easily and rapidly accessing books, archives and images of various types are now widely recognized by commercial interests and public bodies alike.

Traditional libraries are limited by storage space; digital libraries have the potential to store much more information, simply because digital information requires very little physical space to contain it. As such, the cost of maintaining a digital library is much lower than that of a traditional library.

A traditional library must spend large sums of money paying for staff, book maintenance, rent, and additional books. Digital libraries may reduce or, in some instances, do away with these fees. Both types of library require cataloguing input to allow users to locate and retrieve material. Digital libraries may be more willing to adopt innovations in technology providing users with improvements in electronic and audio book technology as well as presenting new forms of communication such as blogs; conventional libraries may consider that providing online access to their OPAC catalogue is sufficient. An important advantage to digital conversion is increased accessibility to users. They also increase availability to individuals who may not be traditional patrons of a library, due to geographic location or organizational affiliation.

- **No physical boundary**. The user of a digital library need not to go to the library physically; people from all over the world can gain access to the same information, as long as an Internet connection is available.
- **Round the clock availability** A major advantage of digital libraries is that people can gain access 24/7 to the information.

- **Multiple access**. The same resources can be used simultaneously by a number of institutions and patrons. This may not be the case for copyrighted material: a library may have a license for "lending out" only one copy at a time; this is achieved with a system of digital rights management where a resource can become inaccessible after expiration of the lending period or after the lender chooses to make it inaccessible (equivalent to returning the resource).
- **Information retrieval**. The user is able to use any search term (word, phrase, title, name, subject) to search the entire collection. Digital libraries can provide very user-friendly interfaces, giving clickable access to its resources.
- **Preservation and conservation**. Digitization is not a long-term preservation solution for physical collections, but does succeed in providing access copies for materials that would otherwise fall to degradation from repeated use. Digitized collections and born-digital objects pose many preservation and conservation concerns that analog materials do not. Please see the following "Problems" section of this page for examples.
- **Space**. Whereas traditional libraries are limited by storage space, digital libraries have the potential to store much more information, simply because digital information requires very little physical space to contain them and media storage technologies are more affordable than ever before.
- **Added value**. Certain characteristics of objects, primarily the quality of images, may be improved. Digitization can enhance legibility and remove visible flaws such as stains and discoloration.
- **Easily accessible**.

Challenges

Digital preservation

Digital preservation aims to ensure that digital media and information systems are still interpretable into the indefinite future. Each necessary component of the must be migrated, preserved or emulated. Typically lower levels of systems (floppy disks for example) are emulated, bit-streams (the actual files stored in the disks) are preserved and operating systems are emulated as a virtual machine. Only where the meaning and content of digital media and information systems are well understood is migration possible, as is the case for office documents.

Copyright and licensing

Some people have criticized that digital libraries are hampered by copyright law, because works cannot be shared over different periods of time in the manner of a traditional library. The republication of material on the Web by libraries may require permission from rights holders, and there is a conflict of interest between them and publishers who may wish to create online versions of their acquired content for commercial purposes.

There is a dilution of responsibility that occurs as a result of the spread-out nature of digital resources. Complex intellectual property matters may become involved since digital material is not always owned by a library. The content is, in many cases, public domain or self-generated content only. Some digital libraries, such as Project Gutenberg, work to digitize out-of-copyright works and make them freely available to the public. An estimate of the number of distinct books still existent in library catalogues from 2000BC to 1960, has been made.

The Fair Use Provisions (17 USC § 107) under copyright law provide specific guidelines under which circumstances libraries are allowed to copy digital resources. Four factors that constitute fair use are purpose of use, nature of the work, market impact, and amount or substantiality used.

Some digital libraries acquire a license to "lend out" their resources. This may involve the restriction of lending out only one copy at a time for each license, and applying a system of digital rights management for this purpose.

Metadata creation

In traditional libraries, the ability to find works of interest was directly related to how well they were catalogued. While cataloguing electronic works digitized from a library's existing holding may be as simple as copying moving a record for the print to the electronic item, with complex and born-digital works requiring substantially more effort. To handle the growing volume of electronic publications, new tools and technologies have to be designed to allow effective automated semantic classification and searching. While full text search can be used for some searches, there are many common catalog searches which cannot be performed using full text, including:

- finding texts which are translations of other texts
- linking texts published under pseudonyms to the real authors (Samuel Clemens and Mark Twain, for example)
- differentiating non-fiction from parody (The Onion from The New York Times, for example)

Chapter 7

Accounting Information System

An **accounting information system** (**AIS**) is a system.

Accounting information systems are composed of six main components:

1. People: users who operate on the systems
2. Procedures and instructions: processes involved in collecting, managing and storing the data
3. Data: data that is related to the organization and its business processes
4. Software: application that processes the data
5. Information technology infrastructure: the actual physical devices and systems that allows the AIS to operate and perform its functions
6. Internal controls and security measures: what is implemented to safeguard the data

History

Initially, accounting information systems were predominantly developed "in-house" as legacy systems. Such solutions were difficult to develop and expensive to maintain. Today, accounting information systems are more commonly sold as prebuilt software packages from vendors such as Microsoft, Sage Group, SAP and Oracle where it is configured and customized to match the organization's business processes. As the need for connectivity and consolidation between other business systems increased, accounting information systems were merged with larger, more centralized systems known as enterprise resource planning (ERP). Before, with separate applications to manage different business functions, organizations had to develop complex interfaces for the systems to communicate with each other. In ERP, a system such as accounting information system is built as a module integrated into a suite of applications that can include manufacturing, supply chain, human resources. These modules are integrated together and are able to access the same data and execute complex business processes.

With the ubiquity of ERP for businesses, the term “accounting information system” has become much less about pure accounting (financial or managerial) and more about tracking processes across all domains of business.

Software architecture of a modern AIS

A modern AIS typically follows a multitier architecture separating the presentation to the user, application processing and data management in distinct layers. The presentation layer manages how the information is displayed to and viewed by functional users of the system (through mobile devices, web browsers or client application). The entire system is backed by a centralized database that stores all of the data. This can include transactional data generated from the core business processes (purchasing, inventory, accounting) or static, master data that is referenced when processing data (employee and customer account records and configuration settings). As transaction occur, the data is collected from the business events and stored into the system’s database where it can be retrieved and processed into information that is useful for making decisions. The application layer retrieves the raw data held in the database layer, processes it based on the configured business logic and passes it onto the presentation layer to display to the users. For example, consider the accounts payable department when processing an invoice. With an accounting information system, an accounts payable clerk enters the invoice, provided by a vendor, into the system where it is then stored in the database. When goods from the vendor are received, a receipt is created and also entered into the AIS. Before the accounts payable department pays the vendor, the system’s application processing tier performs a three-way matching where it automatically matches the amounts on the invoice against the amounts on the receipt and the initial purchase order. Once the match is complete, an email is sent to an accounts payable manager for approval. From here a voucher can be created and the vendor can ultimately be paid.

Advantages and implications of AIS

A big advantage of computer-based accounting information systems is that they automate and streamline reporting. Reporting is major tool for organizations to accurately see summarized, timely information used for decision-making and financial reporting. The accounting information system pulls data from the centralized database, processes and transforms it and ultimately generates a summary of that data as information that can now be easily consumed and analyzed by business analysts, managers or other decision makers. These systems must ensure that the reports are timely so that decision-makers are not acting on old, irrelevant information and, rather, able to act quickly and effectively based on report results. Consolidation is one of the greatest hallmarks of reporting as people do not have to look through an enormous number of transactions. For instance, at the end of the month, a financial accountant consolidates all the paid vouchers by running a report on the system. The system’s application layer retrieves the data from the database and provides a report with the total amount paid to its vendors for that particular month. With large corporations that generate large volumes of transactional data, running reports with even an AIS can take days or even weeks.

After the wave of corporate scandals from large companies such as Tyco International, Enron and WorldCom, major emphasis was put on enforcing public companies to implement strong internal controls into their transaction-based systems. This was made into law with the passage of the Sarbanes Oxley Act of 2002 which stipulated that companies must generate an internal control report stating who is responsible for an organization's internal control structure and outlines the overall effectiveness of these controls. Since most of these scandals were rooted in the companies' accounting practices, much of the emphasis of Sarbanes Oxley was put on computer-based accounting information systems. Today, AIS vendors tout their governance, risk management, and compliance features to ensure business processes are robust and protected and the organization's assets (including data) are secured.

How to effectively implement AIS

As stated above,accounting information systems are composed of six main components:

1. People: users who operate on the systems
2. Procedures and instructions: processes involved in collecting, managing and storing the data
3. Data: data that is related to the organization and its business processes
4. Software: application that processes the data
5. Information technology infrastructure: the actual physical devices and systems that allows the AIS to operate and perform its functions
6. Internal controls and security measures: what is implemented to safeguard the data

When an AIS is initially implemented or converted from an existing system, organizations sometimes make the mistake of not considering each of these six components and treating them equally in the implementation process. This results in a system being "built 3 times" rather than once because the initial system is not designed to meet the needs of the organization, the organization then tries to get the system to work, and ultimately, the organization begins again, following the appropriate process.

Following a proven process that works, as follows, results in optimal deployment time, the least amount of frustration, and overall success. Most organizations, even larger ones, hire outside consultants, either from the software publisher or consultants who understand the organization and who work to help the organization select and implement the ideal configuration, taking all components into consideration. Certified Public Accountants (CPAs) with careers dedicated to information systems work with small and large companies to implement accounting information systems that follow a proven process. Many of these CPAs also hold a certificate that is awarded by the American Institute of CPAs--the Certified Information Technology Professional (CITP). CITPs often serve as co-project managers with an organization's project manager representing the information technology department. In smaller organizations, a co-project manager may be an outsourced information technology specialist who manages the implementation of the information technology infrastructure.

The steps necessary to implement a successful accounting information system are as follows:

Detailed Requirements Analysis where all individuals involved in the system are interviewed. The current system is thoroughly understood, including problems, and complete documentation of the current system—transactions, reports, and questions that need to be answered are gathered. What the users need that is not in the current system is outlined and documented. Users include everyone, from top management to data entry. The requirements analysis not only provides the developer with the specific needs, it also helps users accept the change. Users who have the opportunity to ask questions and provide input are much more confident and receptive of the change, than those who sit back and don't express their concerns.

Systems Design (synthesis)—The analysis is thoroughly reviewed and a new system is created. The system that surrounds the system is often the most important. What data needs to go into the system and how is this going to be handled? What information needs to come out of the system, and how is it going to be formatted? If we know what needs to come out, we know what we need to put into the system, and the program we select will need to appropriately handle the process. The system is built with control files, sample master records, and the ability to perform processes on a test basis. The system is designed to include appropriate internal controls and to provide management with the information needed to make decisions. It is a goal of an accounting information system to provide information that is relevant, meaningful, reliable, useful, and current. To achieve this, the system is designed so that transactions are entered as the occur (either manually or electronically) and information is immediately available on-line for management to use.

Once the system is designed, an RFP is created detailing the requirements and fundamental design. Vendors are asked to respond to the proposal and to provide demonstrations of the product and to specifically respond to the needs of the organization. Ideally, the vendor will input control files, sample master records, and be able to show how various transactions are processed that result in the information that management needs to make decisions. An RFP for the information technology infrastructure follows the selection of the software product because the software product generally has specific requirements for infrastructure. Sometimes, the software and the infrastructure is selected from the same vendor. If not, the organization must ensure that both vendors will work together without "pointing fingers" when there is an issue with either the software or the infrastructure.

Documentation—As the system is being designed, it is documented. The documentation includes vendor documentation of the system and, more importantly, the procedures, or detailed instructions that help users handle each process specific to the organization. Most documentation and procedures are on-line and it is helpful if organizations can add to the help instructions provided by the software vendor. Documentation and procedures tend to be an afterthought, but is the insurance policy and the tool that is used during testing and training—prior to launch. The documentation is tested during the training so that when the system is launched, there is no question that it works and that the users are confident with the change.

Testing—Prior to launch, all processes are tested from input through output, using the documentation as a tool to ensure that all processes are thoroughly documented and that users can easily follow the procedures so that you know it works and that the procedures will be followed consistently by all users. The reports are reviewed and verified, so that there's not a garbage in-garbage out. This is all done in a test system not yet fully populated with live data. Unfortunately, most organizations launch systems prior to thorough testing, adding to the end-user frustration when processes don't work. The documentation and procedures may be modified during this process. All identified transactions must be tested during this step in the process. All reports and on-line information must be verified and traced through the "audit trail" so that management is ensured that transactions will be handled consistently and that the information can be relied upon to make decisions.

Training—Prior to launch, all users need to be trained, with procedures. This means, a trainer using the procedures to show each end user how to handle a procedures. The procedures often need to be updated during training as users describe their unique circumstances and the "design" is modified with this additional information. The end user then performs the procedure with the trainer and the documentation. The end user then performs the procedure with the documentation alone. The end-user is then on his or her own with the support, either in person or by phone, of the trainer or other support person. **This is prior to data conversion.**

Data Conversion—Tools are developed to convert the data from the current system (which was documented in the requirements analysis) to the new system. The data is mapped from one system to the other and datafiles are created that will work with the tools that are developed. The conversion is thoroughly tested and verified prior to final conversion. Of course, there's a backup so that it can be restarted, if necessary.

Launch—The system is implemented only AFTER all of the above is completed. The entire organization is aware of the launch date. Ideally, the current system is retained and oftentimes run in "parallel" until the new system is in full operation and deemed to be working properly. With the current "mass-market" software used by thousands of companies and fundamentally proven to work, the "parallel" run that is mandatory with software tailor-made to a company is generally not done. This is only true, however, when the above process is followed and the system is thoroughly documented and tested and users are trained PRIOR to launch.

Support—The end-users and managers have ongoing support available at all times. System upgrades follow a similar process and all users are thoroughly appraised of changes, upgraded in an efficient manner, and trained.

Many organizations chose to limit the amount of time and money spent on the analysis, design, documentation, and training, and move right into software selection and implementation. It is a proven fact that if a detailed requirements analysis is performed with adequate time being spent on the analysis, that the implementation and ongoing support will be minimal. Organizations who skip the steps necessary to ensure the system meets the needs of the organization are often left with frustrated end users, costly support, and information that is not current or correct. Worse yet, these organizations build the system 3 times instead of once.

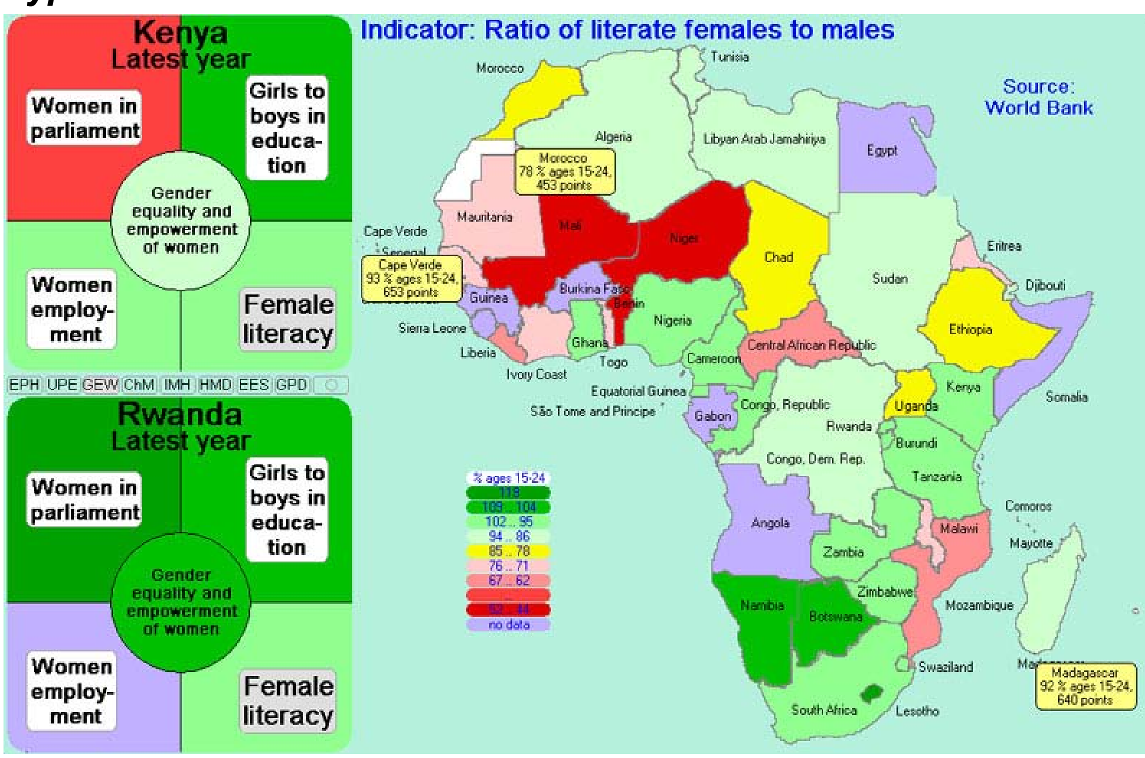
Chapter 8

Dashboards (Management Information Systems) & Risk Management Information Systems

Dashboards (Management Information Systems)

In management information systems, a **dashboard** is an executive information system user interface that (similar to an automobile's dashboard) is designed to be easy to read. For example, a product might obtain information from the local operating system in a computer, from one or more applications that may be running, and from one or more remote sites on the Web and present it as though it all came from the same source. Dashboards should not be confused with scorecards

Types of dashboards



Dashboard of Sustainability screen shot illustrating example dashboard layout.

Digital dashboards may be laid out to track the flows inherent in the business processes that they monitor. Graphically, users may see the high-level processes and then drill down into low level data. This level of detail is often buried deep within the corporate enterprise and otherwise unavailable to the senior executives.

Three main types of digital dashboard dominate the market today: stand alone software applications, web-browser based applications, and desktop applications also known as desktop widgets. The last are driven by a widget engine.

Specialized dashboards may track all corporate functions. Examples include human resources, recruiting, sales, operations, security, information technology, project management, customer relationship management and many more departmental dashboards.

Digital dashboard projects involve business units as the driver and the information technology department as the enabler. The success of digital dashboard projects often depends on the metrics that were chosen for monitoring. Key performance indicators, balanced scorecards, and sales performance figures are some of the content appropriate on business dashboards.

Interface design styles

Like a car's dashboard (or control panel), a software dashboard provides decision makers with the input necessary to "drive" the business. Thus, a graphical user interface may be designed to display summaries, graphics (e.g., bar charts, pie charts, bullet graphs, "sparklines," etc.), and gauges (with colors similar to traffic lights) in a portal-like framework to highlight important information.

History

The idea of digital dashboards followed the study of decision support systems in the 1970s. With the surge of the web in the late 1990s, digital dashboards as we know them today began appearing. Many systems were developed in-house by organizations to consolidate and display data already being gathered in various information systems throughout the organization. Today, digital dashboard technology is available "out-of-the-box" from many software providers. Some companies however continue to do in-house development and maintenance of dashboard applications. For example, GE Aviation has developed a proprietary software/portal called "Digital Cockpit" to monitor the trends in aircraft spare parts business.

In the late 1990s, Microsoft promoted a concept known as the Digital Nervous System and "digital dashboards" were described as being one leg of that concept.

Benefits of digital dashboards

Digital dashboards allow managers to monitor the contribution of the various departments in their organization. To gauge exactly how well an organization is performing overall, digital dashboards allow you to capture and report specific data points from each department within the organization, thus providing a "snapshot" of performance.

Benefits of using digital dashboards include:

- Visual presentation of performance measures
- Ability to identify and correct negative trends
- Measure efficiencies/inefficiencies
- Ability to generate detailed reports showing new trends
- Ability to make more informed decisions based on collected business intelligence
- Align strategies and organizational goals
- Saves time compared to running multiple reports
- Gain total visibility of all systems instantly
- Quick identification of data outliers and correlations

Risk Management Information Systems

Risk Management Information Systems (RMIS) are typically computerized systems that assist in consolidating property values, claims, policy, and exposure information and provide the tracking and management reporting capabilities to enable you to monitor and control your overall cost of risk.

General Overview

The management of risk data and information is key to the success of any risk management effort regardless of an organization's size or industry sector. Risk management information systems/services (RMIS) are used to support expert advice and cost-effective information management solutions around key processes such as:

- Risk identification and assessment
- Risk control
- Risk financing

Typically, RMIS facilitates the consolidation of insurance related information, such as claims from multiple sources,property values, policy information, and exposure information, into one system. Often, Risk Management Information Services/Systems (RMIS) applies primarily to “casualty” claims/loss data systems. Such casualty coverages include Auto Liability, Auto Physical Damage, Workers' Compensation, General Liability and Products Liability.

RMIS products are designed to provide their insured organizations and their brokers with basic policy and claim information via electronic access, and most recently, via the Internet. This information is essential for managing individual claims, identifying trends, marketing an insurance program, loss forecasting, actuarial studies and internal loss data communication within a client organization. They may also provide the tracking and management reporting capabilities to enable one to monitor and control overall cost of risk in an efficient and cost-effective manner.

In the context of the acronym RMIS, the word "risk" pertains to an insured or self-insured organization. This is important because prior to the advent of RMIS, insurance company loss information reporting typically organized loss data around insurance policy numbers. The historical focus on insurance policies detracted from a clear, coherent and consolidated picture of a single customer's loss experience. The advent of RMIS in the 1980s was a breakthrough step in the insurance industry's evolution toward persistent and focused understanding of their end-customer needs. Typically, the best solution for your organization depends on whether it is enhancing an existing RMIS system, ensuring the highest level of data quality, or designing and implementing a new system while maintaining a focus on state-of-the-art technology.

Common Types of RMIS

Most major insurance companies (carriers), broker/agents, and Third Party Administrators (TPAs)offer/provide at least one external RMIS product to their insureds (clients) and any brokers involved in the insurance program. Most commonly, RMIS products allow individual claim detail look-up, basic trend report production, policy summaries and ad hoc queries. The resulting information can then be shared throughout the client's organization, usually for insurance program cost allocation, loss prevention and effective claim management at the local level. More advanced products allow multiple claim data sources to be consolidated into one "Master RMIS," which is essential for most large client organizations with complex insurance programs.

The primary users of RMIS are risk/insurance departments of insured organizations and any insurance broker involved. Interestingly, it is much less common for the insured's safety department and vehicle operations department to have access to RMIS despite similar interest in the data. In fact, safety and vehicle operations of larger organizations typically maintain their own separate database systems of "accidents/incidents," many of which will correlate to RMIS claim data.

Insurance companies normally use a different version of externally provided RMIS for internal use, such as by underwriting and loss control personnel. Occasionally, there could be timing or other differences that could cause data discrepancies between the internal system and externally provided RMIS.

Insurance brokers have a similar need for access to their insured client's claim data. Brokers are normally added as an additional user to the RMIS product provided to their clients by the insurance carrier and TPAs. The information available from RMIS is critical to the broker for interfacing effectively with their counterparts in the insurance carrier and TPAs. Additionally, effectively presented RMIS information that shows trends and analysis is essential to successfully marketing their clients' insurance programs.

Insurance carrier and Third-Party Administer (TPA) claim adjusters traditionally use claims management systems to collect and manage claim information and to administer claims. Some client organizations, however, may choose to manage certain types of claims or those within a loss retention layer and thus use this type of system as well.

Typically, the claims management system provides the primary data to RMIS products. RMIS products in turn provide an externally accessed view into the client's claims data. RMIS products are commonly available directly from larger insurance carriers and TPAs, but the most advanced systems are often offered by independent RMIS vendors. Independent RMIS vendor systems are most desirable when a client organization needs to consolidate claims data from multiple current insurance programs and/or past programs with current program information.

Key Vendor Attributes and Differences

Along with insurance carriers, broker/agents and TPAs that offer their own proprietary systems, there are a variety of direct RMIS technology companies who sell to direct insureds and even the carriers, broker/agents and TPAs themselves.

Major differences among RMIS vendors include:

- Currency of technology (Internet-based vs. Internet-accessible);
- System speed (response time for screen changes, report generation time, etc.);
- Flexibility in meeting client requirements (custom screen views, client-defined data fields, special reports, etc.);
- Ongoing support service quality (availability of senior/quality technical support, help desk availability, dedicated staff and stability, etc.);
- Data quality control (data conversion accuracy, data source cleanup, etc.);
- Pricing (first-year cost, ongoing cost, custom programming charges, data record storage fees);
- Availability of related modules (property exposure management, policy management, claim/incident setup, Occupational Safety and Health Administration (OSHA) record keeping, claims audits, etc.);
- Turnaround time for data loads;
- Foreign conversion/support (financial fields, language, fluent support staff, etc.)

RMIS system compatibility varies among carriers, broker/agents and TPAs. However, quality independent RMIS vendors by design can take almost any claim data source and convert or map the data to their particular system's file structure. A few major insurance carriers offer similar consolidation services, i.e., combining the insured client's current claim data with another carrier's or TPA's data for the same insured client. The other data sources can be for current separate insurance programs or from expired insurance programs. Usually, this type of consolidation service is performed to accommodate their major policyholder organizations. Major TPAs, however, more commonly offer such data consolidation services.

Average RMIS Costs and RMIS Market Drivers

The cost of a typical independent RMIS product varies from $60,000 to $150,000 for the first year, and ongoing annual charges are slightly less. Insurance company RMIS product lines typically average around $50,000 for the first user, but they often offer less expensive light-weight versions for claim look-up only. More costly full-featured products are sometimes available with more advanced reporting systems. The products are usually priced on a per-user basis on a sliding scale for a larger number of users. Insured clients' brokers are given access at no cost or occasionally for a flat annual fee for multiple insured clients with a particular broker.

TPAs commonly include one or two RMIS access IDs within their claims management pricing to encourage both the client's broker and the client to use their claim look-up product. Normally, beyond the first two access IDs, the pricing follows the same per-user range of the insurance companies. The cost drivers of RMIS include: • Number of user/access IDs • Number of outside claim data sources that must be converted (carriers and TPAs do not have to convert their own data) • Frequency of outside claim data updates • Special programming/report development charges • Training of users (initial and annual users' conferences)

Clearly, higher cost systems do not always correlate to better performance in terms of both usefulness and speed. While most carrier and TPA RMIS systems are similarly priced, the independent RMIS vendors' price range varies significantly, as previously mentioned.

Chapter 9

Executive Information System

An **Executive Information System** (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)

The emphasis of EIS is on graphical displays and easy-to-use user interfaces. They offer strong reporting and drill-down capabilities. In general, EIS are enterprise-wide DSS that help top-level executives analyze, compare, and highlight trends in important variables so that they can monitor performance and identify opportunities and problems. EIS and data warehousing technologies are converging in the marketplace.

In recent years, the term EIS has lost popularity in favour of Business Intelligence (with the sub areas of reporting, analytics, and digital dashboards).

History

Traditionally, executive information systems were developed as mainframe computer-based programs. The purpose was to package a company's data and to provide sales performance or market research statistics for decision makers, such as financial officers, marketing directors, and chief executive officers, who were not necessarily well acquainted with computers. The objective was to develop computer applications that would highlight information to satisfy senior executives' needs. Typically, an EIS provides data that would only need to support executive level decisions instead of the data for all the company.

Today, the application of EIS is not only in typical corporate hierarchies, but also at personal computers on a local area network. EIS now cross computer hardware platforms and integrate information stored on mainframes, personal computer systems, and minicomputers. As some client service companies adopt the latest enterprise information systems, employees can use their personal computers to get access to the company's data and decide which data are relevant for their decision makings. This arrangement makes all users able to customize their access to the proper company's data and provide relevant information to both upper and lower levels in companies.

Components

The components of an EIS can typically be classified as:

Hardware

When talking about hardware for an EIS environment, we should focus on the hardware that meet the executive's needs. The executive must be put first and the executive's needs must be defined before the hardware can be selected. The basic computer hardware needed for a typical EIS includes four components:

1. Input data-entry devices. These devices allow the executive to enter, verify, and update data immediately;
2. The central processing unit (CPU), which is the kernel because it controls the other computer system components;
3. Data storage files. The executive can use this part to save useful business information, and this part also help the executive to search historical business information easily;
4. Output devices, which provide a visual or permanent record for the executive to save or read. This device refers to the visual output device such as monitor or printer.

In addition, with the advent of local area networks (LAN), several EIS products for networked workstations became available. These systems require less support and less expensive computer hardware. They also increase access of the EIS information to many more users within a company.

Software

Choosing the appropriate software is vital to design an effective EIS. Therefore, the software components and how they integrate the data into one system are very important. The basic software needed for a typical EIS includes four components:

1. Text base software. The most common form of text are probably documents;
2. Database. Heterogeneous databases residing on a range of vendor-specific and open computer platforms help executives access both internal and external data;
3. Graphic base. Graphics can turn volumes of text and statistics into visual information for executives. Typical graphic types are: time series charts, scatter diagrams, maps, motion graphics, sequence charts, and comparison-oriented graphs (i.e., bar charts);

4. Model base. The EIS models contain routine and special statistical, financial, and other quantitative analysis.

Perhaps a more difficult problem for executives is choosing from a range of highly technical software packages. Ease of use, responsiveness to executives' requests, and price are all reasonable considerations. Further, it should be considered whether the package can run on existing hardware.

User interface

An EIS needs to be efficient to retrieve relevant data for decision makers, so the user interface is very important. Several types of interfaces can be available to the EIS structure, such as scheduled reports, questions/answers, menu driven, command language, natural language, and input/output. It is crucial that the interface must fit the decision maker's decision-making style. If the executive is not comfortable with the information questions/answers style, the EIS will not be fully utilized. The ideal interface for an EIS would be simple to use and highly flexible, providing consistent performance, reflecting the executive's world, and containing help information.

Telecommunication

As decentralizing is becoming the current trend in companies, telecommunications will play a pivotal role in networked information systems. Transmitting data from one place to another has become crucial for establishing a reliable network. In addition, telecommunications within an EIS can accelerate the need for access to distributed data.

Applications

EIS enables executives to find those data according to user-defined criteria and promote information-based insight and understanding. Unlike a traditional management information system presentation, EIS can distinguish between vital and seldom-used data, and track different key critical activities for executives, both which are helpful in evaluating if the company is meeting its corporate objectives. After realizing its advantages, people have applied EIS in many areas, especially, in manufacturing, marketing, and finance areas.

Manufacturing

Basically, manufacturing is the transformation of raw materials into finished goods for sale, or intermediate processes involving the production or finishing of semi-manufactures. It is a large branch of industry and of secondary production. Manufacturing operational control focuses on day-to-day operations, and the central idea of this process is effectiveness and efficiency. To produce meaningful managerial and operational information for controlling manufacturing operations, the executive has to make changes in the decision processes. EIS provides the evaluation of vendors and buyers, the evaluation of purchased materials and parts, and analysis of critical purchasing areas. Therefore, the executive can oversee and review purchasing operations effectively with EIS. In addition, because production planning and control depends heavily on the plant's data base and its communications with all manufacturing work centers, EIS also provides an approach to improve production planning and control.

Marketing

In an organization, marketing executives' role is to create the future. Their main duty is managing available marketing resources to create a more effective future. For this, they need make judgments about risk and uncertainty of a project and its impact on the company in short term and long term. To assist marketing executives in making effective marketing decisions, an EIS can be applied. EIS provides an approach to sales forecasting, which can allow the market executive to compare sales forecast with past sales. EIS also offers an approach to product price, which is found in venture analysis. The market executive can evaluate pricing as related to competition along with the relationship of product quality with price charged. In summary, EIS software package enables marketing executives to manipulate the data by looking for trends, performing audits of the sales data, and calculating totals, averages, changes, variances, or ratios. All of these sales analysis functions help marketing executives to make final decisions.

Financial

A financial analysis is one of the most important steps to companies today. The executive needs to use financial ratios and cash flow analysis to estimate the trends and make capital investment decisions. An EIS is a responsibility-oriented approach that integrates planning or budgeting with control of performance reporting, and it can be extremely helpful to finance executives. Basically, EIS focuses on accountability of financial performance and it recognizes the importance of cost standards and flexible budgeting in developing the quality of information provided for all executive levels. EIS enables executives to focus more on the long-term basis of current year and beyond, which means that the executive not only can manage a sufficient flow to maintain current operations but also can figure out how to expand operations that are contemplated over the coming years. Also, the combination of EIS and EDI environment can help cash managers to review the company's financial structure so that the best method of financing for an accepted capital project can be concluded. In addition, the EIS is a good tool to help the executive to review financial ratios, highlight financial trends and analyze a company's performance and its competitors.

Advantages and disadvantages

Advantages of EIS

- Easy for upper-level executives to use, extensive computer experience is not required in operations
- Provides timely delivery of company summary information
- Information that is provided is better understood

- Filters data for management
- Improves to tracking information
- Offers efficiency to decision makers

Disadvantages of EIS

- System dependent
- Limited functionality, by design
- Information overload for some managers
- Benefits hard to quantify
- High implementation costs
- System may become slow, large, and hard to manage
- Need good internal processes for data management
- May lead to less reliable and less secure data

Future trends

The future of executive info systems will not be bound by mainframe computer systems. This trend allows executives escaping from learning different computer operating systems and substantially decreases the implementation costs for companies. Because utilizing existing software applications lies in this trend, executives will also eliminate the need to learn a new or special language for the EIS package. Future executive information systems will not only provide a system that supports senior executives, but also contain the information needs for middle managers. The future executive information systems will become diverse because of integrating potential new applications and technology into the systems, such as incorporating artificial intelligence (AI) and integrating multimedia characteristics and ISDN technology into an EIS. EIS - timely, efficient and effective in supporting the decision making process.

Chapter 10

Student Information System

A **student info system** (SIS) is a software application for education establishments to manage student data. Student information systems provide capabilities for entering student test and other assessment scores through an electronic grade book, building student schedules, tracking student attendance, and managing many other student-related data needs in a school, college or university. Also known as **student information management system** (SIMS, SIM), **student records system** (SRS), **student management system** (SMS), **campus management system** (CMS) or **school management system** (SMS).

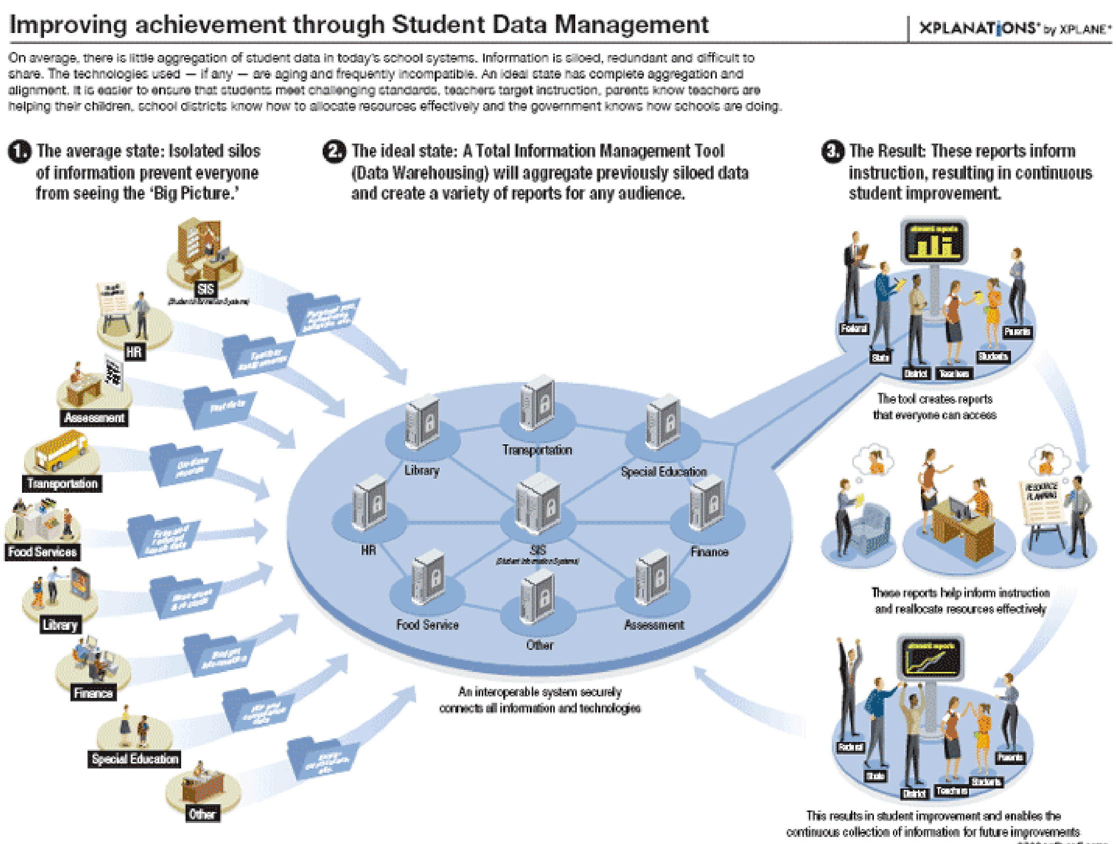


Diagram showing the importance and result of well thought out Student Data Management.

The SIS is equivalent to an Enterprise Resource Planning or ERP system for a corporate customer. As such, many of the issues with ERP System Selection Methodology, implementation, and operation of an ERP system apply to schools and their SIS systems.

Functions

These systems vary in size, scope and capability, from packages that are implemented in relatively small organizations to cover student records alone, to enterprise-wide solutions that aim to cover most aspects of running large multi-campus organizations with significant local responsibility. Many systems can be scaled to different levels of functionality by purchasing add-on "modules" and can typically be configured by their home institutions to meet local needs.

Until recently, the common functions of a student records system are to support the maintenance of personal and study information relating to:

- Handling inquiries from prospective students
- Handling the admissions process
- Enrolling new students and storing teaching option choices
- Automatically creating class & teacher schedules
- Handling records of examinations, assessments, marks, grades and academic progression
- Maintaining records of absences and attendance
- Recording communications with students
- Maintaining discipline records
- Providing statistical reports
- Maintenance boarding house details
- Communicating student details to parents through a parent portal
- Special Education / Individual Education Plan (IEP) services
- Human resources services
- Accounting and budgeting services
- Student health records

In larger enterprise solutions that have student data at their core, further functions include Student financial aid management and more may be customized by the developer. Where national or government systems exist for student finance or statistical return purposes, student records system often provide functionality that caters for this, by way of modules or core elements that handle the production of required files, or deal with the formatted transfer of information. examples are the FAFSA (Free Application for Federal Student Aid) process in the United States, the United Kingdom's Student Loans Company processes (SSAR, SSAC and ATFEE file processing), the UCAS (Universities and Colleges Admissions Service) in the United Kingdom, or the HESA and HESES student statistical returns in the United Kingdom.

In the past, universities and large school districts in particular have created their own bespoke student record systems. One such example is the ROSI system at University of

Toronto. With growing complexity in the business of educational establishments, most organizations now choose to buy customizable software, and increasing numbers are buying software as a service (SAAS). Most student information systems in use today are server-based, with the application residing on a central computer server, and being accessed by client applications at various places within and even outside the school. But student information systems have been moving to the web since the late 1990s and that trend is accelerating as institutions replace older systems.

Integrated Systems / Hosted Service

In recent years, several forces have been driving an evolution of student information systems and, as a result, leading many institutions to replace theirs. Those forces are:

- Demand for 24x7 web-based access to information by students, instructors, and (in primary and secondary education, or K-12) parents
- Increasing demands in the amount and frequency of data reporting for accountability and other purposes (so-called "vertical reporting" up to state, provincial, and national agencies)
- Importance of integrating student information systems with other tools, especially relating to instruction, courses and learning (LMS systems and on-line course ware).
    - SIF (Schools Interoperability Framework) Compliance.

Modern use also implies that smaller K-12 schools can benefit from the reducing cost of technology; this has made it possible for even these organizations to implement such school software that not only encompass the management of student information but also provide the means for parents or guardians to connect with the teaching staff through parent portals

Upgrade Pitfalls

Unlike an upgrade to a web browser or a word processor, changes and upgrades to these systems tend to have significant impact on the day-to-day operations of every school employee. These systems typically touch every aspect of school operations even when only the base modules are used. For these reasons, care should be taken to consider the impact on:

- Workflow: Since these programs are tightly tied to a school's business workflow and processes, a change to a SIS system can force changes to workflow. This can have a significant impact on daily operations if not considered carefully prior to implementation.
- Data Conversion: Data conversion of historic data (transcripts, attendance, health records, etc.) for both current and past students can also be a significant issue to transitioning to a new SIS. Since most schools are required to keep historical data on past students, considerations should be given to what information will be converted and what will be archived.

- Customized Reports: Since there is little standardization in what and how student information is stored, most schools have their own processes and procedures (e.g. formatting and layout of data reports) for student grade printouts or attendance records. As most SISs are not perfectly compatible with the previous SIS, upgrading can be a long, and tedious process.
- Training: Some new SIS programs have a tendency to include some unnecessary features, primary for the use of power users, so training employees to use the new SIS program will most likely be a costly and time-consuming process.

Like with an ERP system, schools should consider processes similar to the ERP System Selection Methodology when selecting a SIS system.

Chapter 11

Laws of Information systems

The **Laws of informations systems** are a collection of observations and generalizations characterizing the behavior of people, hardware, software, and procedures enclosed in a certain scope (an information system).

Introduction

Scientific disciplines such as physics and chemistry are characterized by the existence of a number of laws which define relationships among parameters of interest. Laws help to crystallize the knowledge of a discipline and to extend it to new frontiers. Information systems is an amalgam of scientific and humanistic disciplines including computer science, management science and social sciences. It is therefore characterized by laws as well as a number of principles. These span the range of transaction processing, effective systems, user interfaces and system development.

Laws

The Law of Transaction Volumes The volume of transactions will increase with the stage of development of a society. The greater the development the greater the number of goods and services exchanged and the greater the number of transactions.

The Law of Symbol Systems This is a corollary of the law of transaction volumes. The symbol systems by which messages are encoded grow more complex as a society evolves. Thus the 7 bit Ascii code gave way to the 8 bit version which is giving way to the 16 bit Unicode system. This trend is likely to continue when civilization expands beyond the confines of our planet and the volume of information exchanged becomes astronomical.

The Law of Technological Evolution. Technology seeks the most efficient form, unless otherwise constrained. Efficient form is defined qualitatively as one that is best adapted to its application or as one with the least number of problems. This is a variation on

Darwin's law of evolution and is manifested in the case of memories, storage devices, databases etc.

The Law of Infinite Processing Needs The information processing needs of an organization or society will always exceed its information processing capabilities. This can be seen as a converse of Moore's law – doubling of processing power every two years. Whenever a new technology is introduced it is overwhelmed by new applications or increased usage. For instance, current technologies are not up to the task of processing images streaming from space probes in real time. This is also evidenced in the case of email, internet, search engines etc. that are being swamped by volume.

The Law of Good Systems A good system produces benefits that are disproportionately high in comparison to the initial investment. Any complex system, including an information system is typically interconnected with other systems. So a good system has ripple effects which show up as unexpected benefits. The freeway system in the U.S. for example, led to the growth of the automotive, steel and motel industries. Another example is the Sabre system that has been designed for making reservations but has been used in crew scheduling and flight forecasting. As a corollary a bad system produces problems that are disproportionately high in comparison to its area of operation, a prime example here being the 64K memory limitation of DOS which for a long time stymied software developers.

The Law of Right Design Every software that involves users has a "right" design. The"right design"refers to decomposition of functions into menus/controls. The fact that some types of software are intuitive while others are not leads to the belief that there is a "right design" for every software/IT application that is up to the designer to find.

The Law of Interconnected Systems An interconnected system cannot be controlled unless each interconnect is individually controllable. A very simple example is provided here. In one email system, the "signature" text and message composition are wedded together so that the same signature is produced every time a message is composed. This is also known as "coupling." It provides an option to select the default signature but doesn't allow this to be done dynamically. It should provide an option to select which signature is used, at the time of message composition.

The Law of Complex Interfaces There cannot be a simple interface to a complex system (Here complexity is informally measured as number of menu options). This is a variation on the law of requisite variety which states that variety in a system should be at least as great as that found in its environment. Complex systems such as visual programming environments or CASE tools therefore cannot have simple interfaces.

The Principle of Information Independence Users should be able to access their information regardless of where it is physically located. This is a variation on the concept of distribution independence in databases that has been extended to include other types of information and other types of computing contexts.

The Law of Irregular Transactions Information systems (transaction processing systems) which cannot process irregular transactions are doomed to fail. An irregular transaction is defined as one that deviates from the norm, in terms of items bought, conditions or constraints. Examples include registering for two courses that are scheduled to start at the same time or including a child safety seat in a car rental reservation.

Soft information principle Information systems must incorporate soft information or they are doomed to fail. One way in which irregular transactions can be handled is to provide additional notes on the transaction. Those systems that do not accommodate such soft information may result in a transaction failure or may result in inconveniencing the user/consumer.

The Law of Qualitative Decision Models It is impossible to calculate outcomes with any certainty in a decision situation that involves qualitative variables. This is based on the theory of computability which states that a problem is computable if an algorithm exists, the algorithm is efficient/tractable and if there is a well defined solution state. Because of their very nature, qualitative problems lack well-defined state and hence the law.

The Law of Mental Models The system's model must not exceed the user's mental model in complexity. The system's model is the organization of features in the system whereas the user's mental model is their conceptualization of the system. When the system's model exceeds the user's model, user will not be able to operate the system without extensive training and the result is often an implementation failure.

Principles

The Reapportionment Principle Tasks that can be performed by the system (in the context of software use) should be performed by it. Automatic filling of personal details from ss# or phone# in a customer registration form is one example. This is ultimately based on the simple economic principle of labor substitution to leverage productivity. It's a widely used principle nowadays.

The Principle of Sharing User Information All desktop systems must share information about the user. This is a corollary of the re-apportionment principle. To the extent that desktop systems require user information (such as email address, phone# etc.) it is advantageous for users to have the system obtain it from a common profile.

Information Responsibility Principle Those who have information are obliged to share it with those who need it. This is a principle attributed to Peter Drucker in his1988 HBR article, "The Coming of the Knowledge-Based Organization". Since information is intangible, it is difficult for potential consumers of information to perceive its source and hence the principle. The principle implicitly assumes that reasons for excluding information specifically from a person do not exist.

The Principle of Information Ownership Owners of information must have access to it. This is a corollary of the information responsibility principle. When information changes, the owner has a stake in making the change in the system so it is reasonable to give them the access to do it. Many companies are web-enabling their systems, thereby illustrating this principle.

The Law of Accelerating Returns The rate of progress of technology is accelerating to such an extent that it produces returns that are not linear, but exponential. Much like the Moore's law, every decade there is a doubling of progress and therefore technological advances will occur exponentially. Kurzweil expects that the next great evolutionary step of human kind will be the integration of human biological components with machines.

Brooks Law Adding manpower to a late software project makes it late. Fred Brooks was the chief engineer overseeing the 360 project, which was one of the largest software projects ever undertaken. Based on his experience, Brooks came to the conclusion that putting additional programmers on a delayed project will not hasten its implementation because of the additional communications overhead. In fact, it had the tendency to reduce productivity.