MAKERERE UNIVERSITY BUSINESS SCHOOL
DEPARTMENT OF BUSINESS COMPUTING
SYSTEMS ANALYSIS AND DESIGN

BBC 2- Lecture Notes
By

Prof. Joseph Kizito Bada

Date: January 15, 2024

Information system (IS)

- is an arrangement of people, data, processes, information presentation, and information technology that interact to support and improve day-to-day operations in a business as well as support the problem-solving and decision-making needs of management and users

Information and communication technology (ICT)

- is a contemporary term that describes the combination of computer technology (hardware and software) with telecommunications technology (data, image and voice networks).

Components of an Information system

The figure below illustrates an information systems model that expresses a fundamental conceptual framework for the major components and activities of information systems. An information system depends on resources of people (end users and IS specialists), hardware (machines and media), software (programs and procedures), data (data and knowledge bases), and networks (communications media and network support) to perform input, processing, output, storage, and control activities that convert data resources into information products.

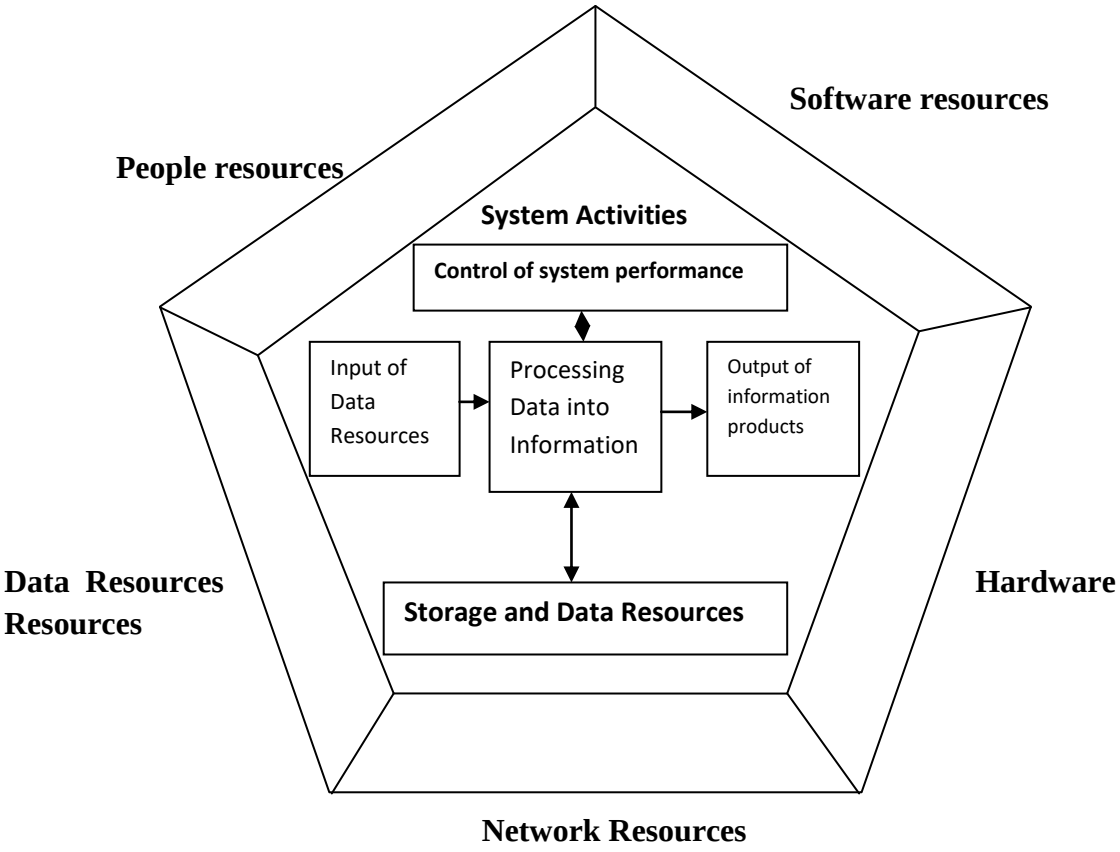


Figure 1: The components of an information system. All information systems use people, hardware, software, data and network resources to perform input, processing, output, storage and control activities that transform data resources into information products.

Information System Resources and Products
People Resources **Specialists** – systems analysts, software developers, systems operators **End users** – anyone else who uses information systems
Hardware Resources **Machines** – computers, video monitors, magnetic disk drives, printers, optical scanners. **Media** – floppy disks, magnetic tape, optical disks, plastic cards, paper forms
Software Resources **Programs** – operating system programs, spreadsheet programs, word processing programs, payroll programs. **Procedures** – data entry procedures, error correction procedures, paycheck distribution procedures
Data Resources Product descriptions, customer records, employee files, inventory databases
Network Resources Communications media, communications processors, network access and control software
Information Products Management reports and business documents using text and graphics displays, audio responses and paper forms.

Information systems Activities

Input of data resources – data about business transactions and other events must be captured and prepared for processing by input activity. Input typically takes the form of data entry activities such as recording and editing.

Processing of data into information – data are typically subjected to processing activities, such as calculating, comparing, sorting, classifying, and summarizing. These activities organize, analyze and manipulate data, thus converting them into information for end users.

Output of information products – information in various forms is transmitted to end users and made available to them in the output activity. The goal of information systems is the production of appropriate information products for end users. Common information products include messages, reports, forms and graphic images, which may be provided by video displays, audio responses, paper products and multimedia.

Storage of data resources – storage is a basic system component of information systems. Storage is the information systems activity in which data are retained in an organized manner for later use.
Control of system performance – An information system should produce feedback about its input, processing, storage and output activities. This feedback must be monitored and evaluated to determine if the system is meeting established performance standards.

Why Study Systems Analysis and Design Methods?

1) Today, it is hard to imagine any industry or business that has not been affected by computer based information system and computer applications. Many businesses consider management of their information resources to be equal in importance to managing their other key resources: Property, facilities, equipment, employees and capital
2) Projects do not succeed by chance successful IT projects follow systematic "Analysis and Design" Process
3) In Developing countries, there are additional causes of IT project failures:
    a. Perception that Hardware alone is the computerized information system.
    b. High level of corruption
    c. Fear of Technology

Stakeholders in Information Systems Development
The stakeholders for an information system can be broadly classified into six broad categories:

1. **System owners** – pay for the system to be built and maintained. They own the system, set priorities for the system, and determine policies for its use. In some cases, system owners may also be system users.
2. **System users** – actually use the system to perform or support the work to be completed. System users define the business requirements and performance expectations for the system to be built.
3. **System designers** – design the system to meet the users' requirements. In many cases these technical people may also be the system builders.
4. **System builders** – construct, test and deliver the system into operation.
5. **Systems Analysts** – facilitate the development of information systems and computer applications by bridging the communications gap that exists between the non-technical system owners and users and technical system designers and builders.

6. **IT vendors and consultants** – sell hardware, software, and services to businesses for incorporation into their information systems.

The Modern Systems Analyst

A systems analyst studies the problems and needs of an organization to determine how people, data, processes, communications and information technology can best accomplish information for the business.

A business analyst is a systems analyst that specializes in business problem analysis and technology-independent requirements analysis.

In information systems development, the term problem is used to describe the following situations:

1) True problem situation either real or anticipated, that require corrective action
2) Opportunities to improve a situation despite the absence of complaints.
3) Directives to change situations regardless of whether anyone has complained.

Duties of a systems analyst

Most systems analysts use some variation of a problem-solving approach, which usually incorporates the following general steps:

1. Identify the problem
2. Analyze and understand the problem
3. Identify solution requirements and expectations
4. Identify alternative solutions and decide a course of action
5. Design and implement the best solution
6. Evaluate the results, if the problem is not solved, return to step 1 or 2 as appropriate.

Qualities of systems analyst

1. **Working knowledge of information technology** – the systems analyst is an agent of change. He or she is responsible for showing end-users and management how new technologies can benefit their business and operations.
2. **Computer programming experience and expertise** – whether or not systems analysts write programs, they must know how to program because they are the principal link between business users and computer programmers.
3. **General business knowledge** – (Accounting, business law and ethics, economics, finance, manufacturing, marketing, operations management,

organizational behavior), systems analyst should be able to communicate with business experts to gain knowledge of problems and needs.

4. **Problem-solving skills** – a systems analyst should be able to take a large business problem, break down that problem into its component parts, analyze the various aspects of the problem, and then assemble an improved system to solve the problem. Engineers call this problem-solving process *analysis and synthesis*. The analyst must learn to analyze problems in terms of causes and effects rather than in terms of simple remedies.
5. **Interpersonal communication skills** – without exception, an analyst must be able to communicate effectively both orally and in writing. An analyst should be skilled in business writing, technical writing, interviewing, presentations and listening.
6. **Interpersonal relations skills** - As a facilitator of information systems development, the systems analyst needs to exercise boldness, and common sense. System work is people-oriented and the systems analyst must be extroverted. Interpersonal skills help the systems analyst to work effectively with people.
7. **Flexibility and adaptability** – no two systems development projects encountered by a systems analyst are identical. Each project offers its unique challenges. Successful systems analysts recognize this and learn to be flexible and adapt to special challenges and situations.
8. **Character and Ethics** – the nature of the systems analysts job requires strong character and sense of ethics. Ethics is a personal character trait in which an individual understands the difference between "right" and "wrong" and acts accordingly. Systems analysts often encounter sensitive information when developing systems. It could be a file of company's pricing structure for a bid or employee profile data that may include salaries, performance history, medical history, and career plans. Analysts must protect the security and confidentiality of any data they have been entrusted with. Trust is sacred! Confidence is earned!

The need for systems analysis and design

At first, the following might seem a plausible course of action by an organization when purchasing and installing a computer system. The first step is to identify the application areas. For instance, these might be accounting, budgeting and word processing. A search is then made of the computer literature in order to establish the names and reviews of accounting, spreadsheet and word-processing packages. A small group of likely candidates for each application is selected. These are then

demonstrated by the dealers selling them and the package that best meets the needs of the users for such application is chosen. Compatible hardware is then purchased, often recommended by the dealers, and the equipment is installed, the software and existing business data are loaded up and, hey presto! The organization has a working information system that meets the requirements of the users and delivers all the benefits associated with computerization.

This approach may work when a small business is in need of computer-assisted support for its standard procedures and when those needs are clearly identified. It is unlikely that it will be satisfactory for the development of a more complex system for a medium-sized or large organization. As a role, the larger the organization the more complex and individual are the data-processing and information needs of that organization and the greater is the potential amount of funding available for a computer project. These organizations are most likely to develop their own system or pay specialist firms to do this for them. Their needs are individual and often initially unidentified. Custom designed systems are required.

For larger systems, the requirements users must be identified and suitable system designed and specified meeting those needs. This must take account of the hardware, the software and the data storage structure. The design must incorporate control and security features. It must also take account of predicted future needs. The hardware will be purchased and installed, the programs, written and tested and existing data loaded into the data structure. Networking infrastructure may need to be installed and adequate security arrangements with external networks established. The system will be tested and necessary amendments made. Staff training will be organized. Finally, after the system is up and running, continued maintenance will be necessary. All of this requires many people with differing areas of expertise. The sums of money involved may be large, the time taken for completion many months or even years. It is essential that the project is planned and coordinated properly.

The need for an information systems strategy

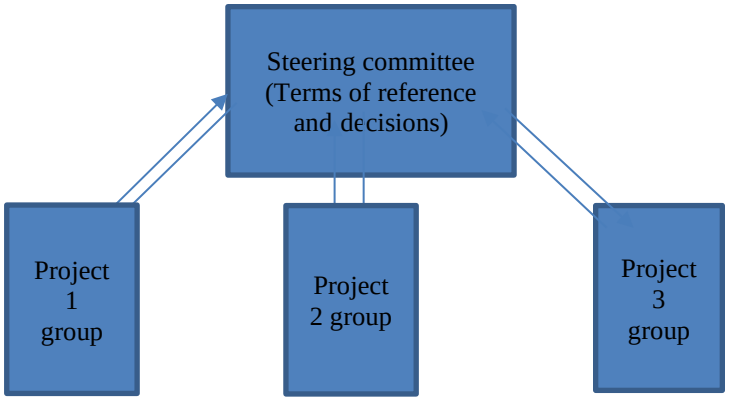
During the past 40 years computerized data processing and information provision have changed vastly. Developments in technology have included the microprocessor, sophisticated telecommunications systems, networking and internet, new office automation equipment and development of cheap, user-friendly packaged software. These changes have allowed cheap and powerful processing facilities to be open to all parts of an organization. Within the organization, the needs of users have evolved rapidly. In order to prevent anarchic

chaos through the development of many independent internal information systems, it is necessary to provide some kind of control by way of a well-worked-out information systems strategy.
This strategy will aim to identify those business activities within the organization that are appropriate to computerized systems development. It will map out, in broad terms, a plan for the development of projects. The strategy will look closely at the size of the investment and consider which of the returns are appropriate and where they will come from. It will incorporate new developments in technology and future needs wherever possible. This strategy will also decide between a policy of centrally controlled development of projects and a strategy of local developments to meet local needs. This latter approach is likely to be an applicable to large organizations that already have a philosophy of dispersed management control.

Information systems steering committees

The responsibility for over strategic planning and control of computer systems development will usually reside with a standing steering committee. This will not be a committee required to take detailed technical decisions. In fact, many of its members will have little technical knowledge or experience. It will be required to frame overall development strategies and allocate resources. Its aim is to ensure that the information systems in the organization deliver an effective service compatible with cost efficiency.

Figure 2 The relationship between the steering committee and project groups



The purposes of the committee may lie within the following areas:

- **To recommend an overall policy information and data-processing systems development**:

This will include such issues as whether or not to standardize on the computer equipment of one company, whether to go for a centralized or decentralized systems, the method of charging for development and use of computer systems within the company, the policy on data protection legislation, and the resources available for information systems projects.

- To ensure that individual user department needs are being satisfied: the information system should service the needs of the organization. The presence of individual user department managers or representatives ensures that user views are articulated.
- To initiate and monitor individual projects: this will include specification of budgets, the scope and objectives of each project; setting up a project team for each project and determining its items of reference; receiving progress reports and taking major decisions, for example stop/go on a project (see figure 2).
- To coordinate individual projects: individual projects that will affect one another must be made aware of this in order to ensure harmonious development. It is also important that projects are not viewed independently but taken together as a strategy.
- To report 'upwards' to top management: management will need summary reports on project development and present and future costs.
- To be responsible for the appointment of senior personnel in the computer Centre: job specifications and appointments will be decided at this managerial level.

Typically, the steering committee will meet regularly, it will be composed of managers of the departments that use the information systems, the head of the computer Centre of its equivalent and other senior members of the computer center, such as the chief analyst, and other person that senior management judges necessary. One of the most important functions of the steering committee is to initiate and set the terms of reference for new projects.

Reasons for project initiation

The projects are initiated by the steering committee, but where does the idea for a new development come from? What causes recognition of the need for computer systems development? There are a number of reasons, many of which can be related to porter's models of competitive advantage introduced in chapter 2. The following are among the most common:

1. **The current system cannot cope:** many systems projects replace old systems. The previous system may have been a manual system or be based on a computer. Either way it may not have been able to cope with the demands on it. For instance, increases in the volume of transactions processed may have made the system so slow that it ceases to be efficient. Backlogs in orders may build up. Staff may be blogged down with excessive paperwork of a routine nature. Or a merger may lead to a change in organizational structure that renders the current system inappropriate.
2. **Cost savings**: one of the most common reasons for the earliest computerization projects was the replacement of time-consuming and therefore expensive, manual, rule-governed, repetitive procedures by quick, cheap computer substitutes. This was most notable in the area of payroll and mass billing for the nationalized industries, where computer systems quickly and cheaply carried out the tasks of entire rooms of clerical workers. Nowadays, most savings in these areas have been made and this is rarely a reason for computerization.
3. **The provision of better internal information for decision making**: management has recognized the ability of computers to supply fast, accurate, targeted information. If management decisions are analyzed for their information requirements, information systems can be designed to enable more effective decisions to be taken.
4. **The provision of competitive customer services**: This may range from fast enquiry services and clear, itemized bills to customer-operated input/output equipment. Automatic cashpoint systems are in the latter category. Once one bank supplies this service they all must or else lose their customers.
5. **The opportunities provided by new technology**: Unlike cars, old computer systems are rarely scrapped because they wear out. It is possible though that outdated technology does not offer the same range of facilities as that currently being produced. Networks, improvements in storage devices and processing power, the development of micros with their cheap and end-user oriented packages and the widespread adoption and application of the internet have in themselves opened new doors for the exploitation of the benefits of computerization.
6. **High-technology image**: some companies feel that their image suffers unless they are seen to be using computers in their operations. These

companies always display the technology in prominent areas, such as reception.

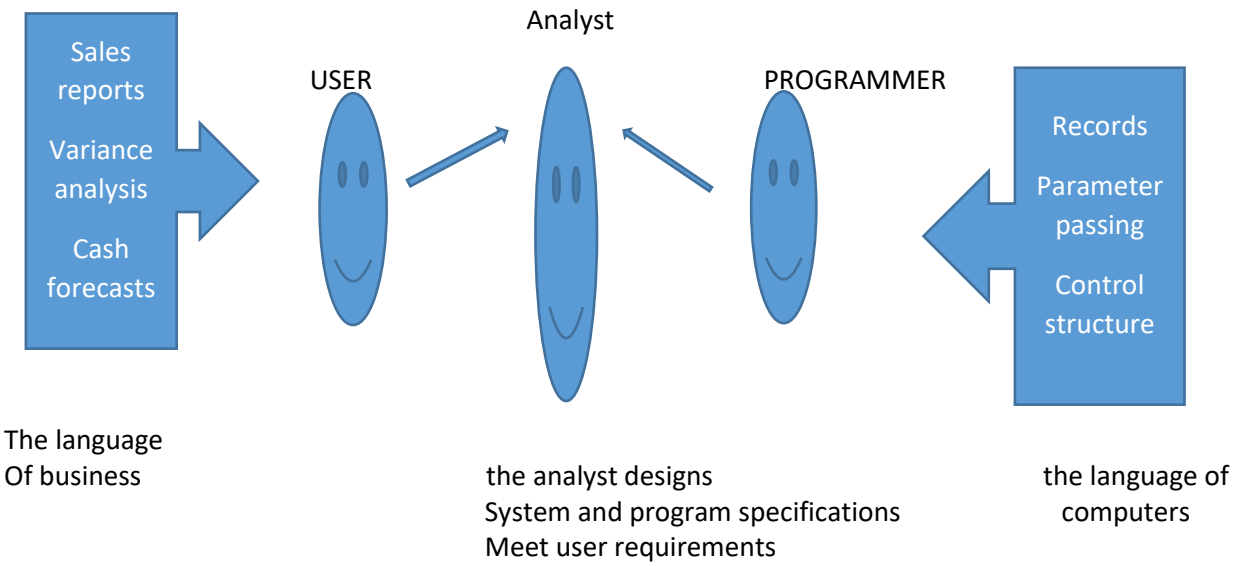
7. **Changes in legislation**: changes in legislation such as data protection legislation may act as the trigger for new systems development. Other examples include significant alterations to taxations to taxation or changes to National Insurance legislation in the UK, or the basis for the preparation of company accounts.
8. **Balancing the portfolio of projects**: an organization may have a policy of spreading the risk inherent in new project developments. This may lead to a range of projects being undertaken, from those with low risk (and possibly low business impact) through to those with high risk (but potentially with business impact).

The participants in analysis and design

It is common for computer systems project to be initiated because someone has recognized that a problem exists with the way that things are currently done. Alternatively, an opportunity is perceived that will lead to an improvement on the present system. In either case, the users of the existing system will play an important role. They can provide information on the current system. They will also be able to specify, in their own terms, the requirements of the new system.

Systems developers or programmers are responsible for turning those requirements into programs. They may either be writing code from scratch, using program generators, or applying system development tools or database management systems. When executed, these will control the operation of the computer, making it perform to serve the needs of users. However, the programmer will be a computer specialist and will see the problem in computer terms. He or she will be talking a different language from the users. There is a communication gap.

Figure 3 the role of the analyst



This gap is filled by the systems analyst or business analyst. This person is able to understand and communicate with users to establish their requirements. The analyst will also have an expert knowledge of computers. He or she will reframe those requirements in terms that programmers can be a good communicator who can think in terms of the user's point of view as well that of the programmer (see figure 10.3).

It is more helpful to think of it of it along the lines of architecture and building. The client (user) states his or her understanding of what the building should look like and what function it should perform (user statement of the requirements of the system). The architect (analyst) then takes these intentions and provides general sketch of the building that will satisfy them (logical model of the intended system). Having agreed this with (user) the architect (analyst) then draws up a detailed blueprint of the design (detailed program specification) from which the builders (programmers) can work. Just as the architect's task is skilled one requiring knowledge of building materials, so the analyst needs knowledge of computers. The analyst's task is not restricted to providing specifications for the programmers. The analyst has a range of responsibilities:

1. The analyst is responsible for investigating, and analyzing the existing system as to its information use and requirements.
2. The analyst judges whether it is feasible to develop a computer system for the area.
3. The analyst designs the new system, specifying programs, hardware, data, structures and control and another procedure.

4. The analyst will be responsible for testing and overseeing the installation of the new system, generating documentation governing its functioning and evaluating its performance.

The analyst is likely to come from either a computer science or a business background. He or she will usually possess a degree and/or be professionally qualified. Sometimes an analyst may have risen 'through the ranks' from programmer to programmer/analyst to a full systems analyst.
As well as possessing significant technical skills in computing, the analyst must fully appreciate the environment and work practices of the area in which the computer system will be used. Knowledge and experience are necessary but not sufficient. The analyst must, above all, be a good communicator with business personnel as well as with technical staff. He or she must be able to handle diplomatically the conflicts of interest that inevitably arise in the course of a project. Managerial, particularly project management, skills are another essential asset as a project involves a complex interaction of many people from different backgrounds working on tasks that all have to be coordinated to produce an end produce an end product. The design process is not mechanical, and the analyst must demonstrate both considerable creativity and the ability to think laterally. Finally, analysts need to exude confidence and controlled enthusiasm. When thing go wrong, it will be the analyst to whom people look as the person to sort out the problems and smooth the way forward.

Factors for the development of good Information Systems

An information system is also a system that accepts resources (data) as input and processes them into products (information) as output. The issues that always ought to be put into consideration when developing an information system include, but are not limited to the following:

1. **Information needs identification**

Managers, field staff, board members and information systems staff must possess a thorough understanding of the information needs of their respective institutions if they are to make relevant decisions of key information systems development issues

2. **Effective communication**

Effective communication between management and systems people is equally important. Financial institutions managers and information systems staff

generally don't speak the same language. Compounding this problem are heavy staff workloads and a tendency to compartmentalize operations. The result, despite the best of intentions, is often a misinterpretation of management requests and a system that does not meet its user's needs.

3. **Susceptibility to Change**

Change management goes hand in hand with consciousness of the realistic expectations about information technology. In this age characterized by dynamism in computer and related information and communications technology, information users often wonder why they can't have the information they want and at the right time they need it. Meeting the aforementioned need seems like a straight forward task- especially when they know that all the necessary data is in a computer

Note:

Management can greatly improve the prospects of developing a good information system through a willingness to evaluate and change the way the institution works. Information systems can perform only as well as the institutions they model. If policies, procedures, job descriptions, workflows and the like haven't been properly established, no MIS will function well. So developing and implementing a new MIS, or reworking an existing one, may affect every part of an institution – and generally should if the process is to be successful.

Systems Development Life Cycle

The *systems development life cycle (SDLC)* is the process of understanding how an information system (IS) can support business needs, designing the system, building it, and delivering it to users.

The key person in the SDLC is the systems analyst who analyzes the business situation, identifies opportunities for improvements, and designs an information system to implement them. Being a systems analyst is one of the most interesting, exciting, and challenging jobs around. As a systems analyst, you will work with a variety of people and learn how they conduct business. Specifically, you will work with a team of systems analysts, programmers, and others on a common mission. You will feel the satisfaction of seeing systems that you designed and developed make a significant business impact, while knowing that you contributed your unique skills to make that happen.

It is important to remember that the primary objective of the systems analyst is not to create a wonderful system. The primary goal is to create value for the organization, which for most companies means increasing profits (government agencies and not-for-profit organizations measure value differently). Many failed systems were abandoned because the analysts tried to build a wonderful system without clearly understanding how the system would fit with the organization's goals, current business processes, and other information systems to provide value. An investment in an information system is like any other investment, such as a new machine tool. The goal is not to acquire the tool, because the tool is simply a means to an end; the goal is to enable the organization to perform work better so it can earn greater profits or serve its constituents more effectively.

The SDLC has a similar set of four fundamental *phases*: planning, analysis, design, and implementation. Different projects may emphasize different parts of the SDLC or approach the SDLC phases in different ways, but all projects have elements of these four phases. Each *phase* is itself composed of a series of *steps,* which rely upon *techniques* that produce *deliverables* (specific documents and files that provide understanding about the project).

Planning

The *planning phase* is the fundamental process of understanding *why* an information system should be built and determining how the project team will go about building it. It has two steps:

1. During *project initiation,* the system's business value to the organization is identified—how will it lower costs or increase revenues? Most ideas for new systems come from outside the IS area (from the marketing department, accounting department, etc.) in the form of a system request. A s*ystem request* presents a brief summary of a business need, and it explains how a system that supports the need will create business value. The IS department works together with the person or department that generated the request (called the *project sponsor*) to conduct a feasibility analysis. The *feasibility analysis* examines key aspects of the proposed project:
    - The technical feasibility (Can we build it?)
    - The economic feasibility (Will it provide business value?)
    - The organizational feasibility (If we build it, will it be used?)

- The system request and feasibility analysis are presented to an information systems *approval committee* (sometimes called a *steering committee*), which decides whether the project should be undertaken.

2. Once the project is approved, it enters—*project management*. During project management, the *project manager* creates a *workplan*, staffs the project, and puts techniques in place to help the project team control and direct the project through the entire SDLC. The deliverable for project management is a *project plan* that describes how the project team will go about developing the system.

Analysis

The *analysis phase* answers the questions of *who* will use the system, *what* the system will do, and *where* and *when* it will be used. During this phase, the project team investigates any current system(s), identifies improvement opportunities, and develops a concept for the new system. This phase has three steps:

1. An *analysis strategy* is developed to guide the project team's efforts. Such a strategy usually includes an analysis of the current system (called the *as-is system*) and its problems, and then ways to design a new system (called the *to-be system*).
2. The next step is *requirements gathering* (e.g., through interviews or questionnaires). The analysis of this information—in conjunction with input from project sponsor and many other people—leads to the development of a concept for a new system. The system concept is then used as a basis to develop a set of business *analysis models* that describes how the business will operate if the new system were developed. The set of models typically includes models that represent the data and processes necessary to support the underlying business process.
3. The analyses, system concept, and models are combined into a document called the *system proposal,* which is presented to the project sponsor and other key decision makers (e.g., members of the approval committee) that decide whether the project should continue to move forward.

The system proposal is the initial deliverable that describes what business requirements the new system should meet.

Design

The *design phase* decides *how* the system will operate, in terms of the hardware, software, and network infrastructure; the user interface, forms and reports; and the specific programs, databases, and files that will be needed. Although most of the strategic decisions about the system were made in the development of the system concept during the analysis phase, the steps in the design phase determine exactly how the system will operate. The design phase has four steps:

1. The *design strategy* must be developed. This clarifies whether the system will be developed by the company's own programmers, whether it will be outsourced to another firm (usually a consulting firm), or whether the company will buy an existing software package.
2. This leads to the development of the basic *architecture design* for the system that describes the hardware, software, and network infrastructure that will be used. In most cases, the system will add or change the infrastructure that already exists in the organization. The *interface design* specifies how the users will move through the system (e.g., navigation methods such as menus and on-screen buttons) and the forms and reports that the system will use.
3. The *database and file specifications* are developed. These define exactly what data will be stored and where they will be stored.
4. The analyst team develops the *program design*, which defines the programs that need to be written and exactly what each program will do.

This collection of deliverables (architecture design, interface design, database and file specifications, and program design) is the *system specification* that is handed to the programming team for implementation. At the end of the design phase, the feasibility analysis and project plan are reexamined and revised, and another decision is made by the project sponsor and approval committee about whether to terminate the project or continue.

Implementation

The final phase in the SDLC is the *implementation phase,* during which the system is actually built (or purchased, in the case of a packaged software design). This is the phase that usually gets the most attention, because for most systems it is longest and most expensive single part of the development process. This phase has three steps:

1. System *construction* is the first step. The system is built and tested to ensure it performs as designed. Since the cost of bugs can be immense, testing is one of the most critical steps in implementation. Most organizations spend more time and attention on testing than on writing the programs in the first place.
2. The system is installed. *Installation* is the process by which the old system is turned off and the new one is turned on. It may include a direct cutover approach (in which the new system immediately replaces the old system), a parallel conversion approach (in which both the old and new systems are operated for a month or two until it is clear that there are no bugs in the new system), or a phased conversion strategy (in which the new system is installed in one part of the organization as an initial trial and then gradually installed in others). One of the most important aspects of conversion is the development of a *training plan* to teach users how to use the new system and help manage the changes caused by the new system.
3. The analyst team establishes a *support plan* for the system. This plan usually includes a formal or informal post-implementation review, as well as a systematic way for identifying major and minor changes needed for the system.

Software Requirements defined

a. A condition or capability needed by a user to solve a problem or achieve an objective.
b. A condition or capability that must be met or possessed by a system or system component to satisfy a contract, standard, specification, or other formally imposed document.

Requirements are…a specification of what should be implemented. They are descriptions of how the system should behave, or of a system property or attribute. They may be a constraint on the development process of the system.

Nowhere more than in the requirements processes do the interests of all the stakeholders in a software or system project intersect. These stakeholders include

- Customers who fund a project or acquire a product to satisfy their organization's business objectives.
- Users who interact directly or indirectly with the product (a subclass of customers).

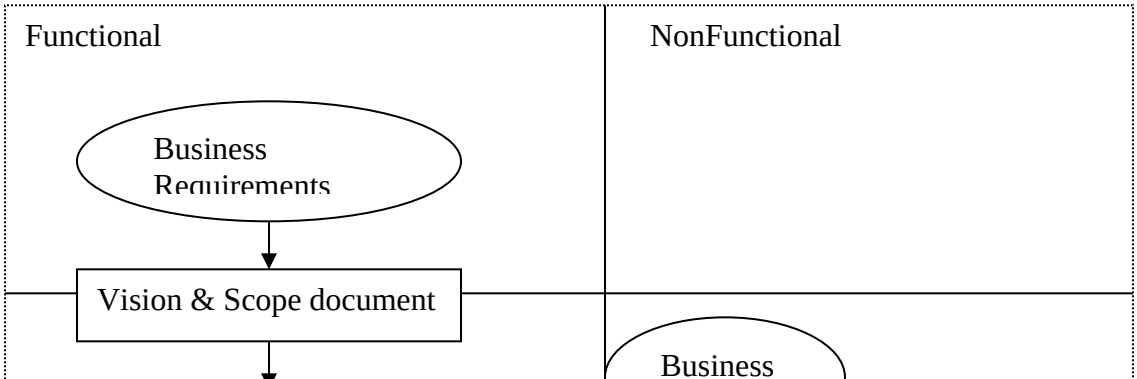
- Requirements analysts who write the requirements and communicate them to the development community.
- Developers who design, implement, and maintain the product.
- Testers who determine whether the product behaves as intended.
- Documentation writers who produce user manuals, training materials, and help systems.
- Project managers who plan the project and guide the development team to a successful delivery.
- Legal staff who ensure that the product complies with all pertinent laws and regulations.
- Manufacturing people who must build the products that contain software.
- Sales, marketing, field support, help desk, and other people who will have to work with the product and its customers.

Handled well, this intersection can lead to exciting products, delighted customers, and fulfilled developers. Handled poorly, it's the source of misunderstanding, frustration, and friction that undermine the product's quality and business value. Because requirements are the foundation for both the software development and the project management activities, all stakeholders must be committed to following an effective requirements process.

Levels of Requirements

Software requirements include three distinct levels—business requirements, user requirements, and functional requirements. In addition, every system has an assortment of nonfunctional requirements. The model in Figure 1 illustrates a way to think about these diverse types of requirements. As with all models, it is not all-inclusive, but it provides a helpful organizing scheme. The ovals represent types of requirements information and the rectangles indicate containers (documents, diagrams, or databases) in which to store that information.

Figure 1. Relationship of several types of requirements information



Business requirements represent high-level objectives of the organization or customer who requests the system. Business requirements typically come from the funding sponsor for a project, the acquiring customer, the manager of the actual users, the marketing department, or a product visionary. Business requirements describe why the organization is implementing the system—the objectives the organization hopes to achieve. Business requirements are recorded in a *vision and scope document*, sometimes called a *project charter* or a *market requirements document*.

User requirements describe user goals or tasks that the users must be able to perform with the product. Valuable ways to represent user requirements include use cases, scenario descriptions, and event-response tables. User requirements therefore describe what the user will be able to do with the system.

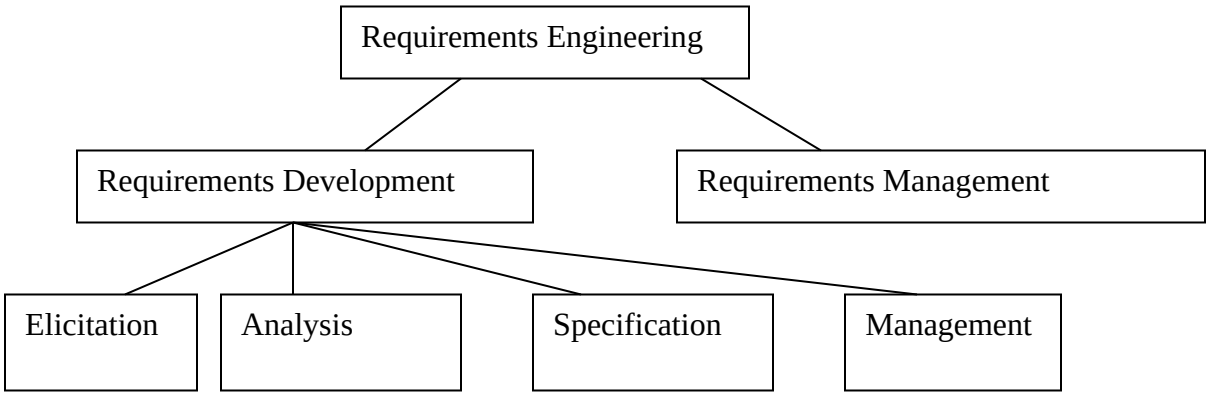
Functional requirements specify the software functionality that the developers must build into the product to enable users to accomplish their tasks, thereby satisfying the business requirements. Sometimes called *behavioral requirements*, these are the traditional "shall" statements: Example, "The system shall e-mail a reservation confirmation to the user."

Business rules include corporate policies, government regulations, industry standards, accounting practices, and computational algorithms. Business rules are not themselves software requirements because they exist outside the boundaries of any specific software system. However, they often restrict who can perform certain use cases or they dictate that the system must contain functionality to comply with the pertinent rules. Sometimes business rules are the origin of specific quality attributes that are implemented in functionality. Therefore, you can trace the genesis of certain functional requirements back to a particular business rule.

Functional requirements are documented in a *software requirements specification* (SRS), which describes as fully as necessary the expected behavior of the software system. The SRS is used in development, testing, quality assurance, project management, and related project functions.

In addition to the functional requirements, the SRS contains nonfunctional requirements. These include performance goals and descriptions of quality attributes. *Quality attributes* augment the description of the product's functionality by describing the product's characteristics in various dimensions that are important either to users or to developers. These characteristics include usability, portability, integrity, efficiency, and robustness. Other nonfunctional requirements describe external interfaces between the system and the outside world, and design and implementation constraints. *Constraints* impose restrictions on the choices available to the developer for design and construction of the product.

Requirements Development and Management



Requirements Development

We can further subdivide requirements development into *elicitation*, *analysis*, *specification*, and *validation*. These subdisciplines encompass all the activities

involved with gathering, evaluating, and documenting the requirements for a software or software-containing product, including the following:

- Identifying the product's expected user classes
- Eliciting needs from individuals who represent each user class
- Understanding user tasks and goals and the business objectives with which those tasks align
- Analyzing the information received from users to distinguish their task goals from functional requirements, nonfunctional requirements, business rules, suggested solutions, and extraneous information
- Allocating portions of the top-level requirements to software components defined in the system architecture
- Understanding the relative importance of quality attributes
- Negotiating implementation priorities
- Translating the collected user needs into written requirements specifications and models
- Reviewing the documented requirements to ensure a common understanding of the users' stated requirements and to correct any problems before the development group accepts them

The PIECES frame work

The PIECES framework focuses on the actual work of doing requirements determination. This model is used to classify identified requirements into one of six subject areas – **P**erformance, **I**nformation, **E**conomy, **C**ontrol, **E**fficiency, and **S**ervices. The goal of the model is to assure the systems analyst and the user that questions will be included during analysis about each of these six essential subjects as it relates to the problem domain. The responses to the questions for each of these subject areas significantly contribute to the definition of the system's requirements.

Performance – questions address how the system needs to perform for the user. Issues of throughput (the amount of work performed over some period of time) and response time (the average delay between a transaction or the user request and the response to that transaction or user request) are considered.

The systems analyst may ask questions about the throughput and the response time for the network, the quality of print needed, the interface required. The major question in this context is "How does the system need to perform in this environment?"

The Information Category – provides the basis for the information or data model that the system needs to maintain. Issues dealing with the input data, output data and stored data are considered.

Economy – this project area addresses economy development and operational cost information along with other objectives that may relate to economy or savings associated with the system.
The Control –category is associated with the system security issues as well as the editing required on the incoming data. Any issues related to controlling the use of the system, its outputs and inputs, or required controls over the data can be included in this category.
Efficiency – is a measure of method correctness. "Are things being done right?" Efficiency's impact is done in at least one of three levels – corporate wide, department, or individual. Questions related to efficiency are primarily directed toward the impact that any solution must have on the environment. For example, how can the operations in the office be improved by this system?
Services – functional requirements of the system. "What does the system need to do in order to solve the problem? "What processes need to be performed?" Ease of use and needed support for ongoing use of the system, maintenance of the system, and training and documentation requirements.

Kozar's Requirements Model
Kozar's requirements model is the third framework. It too focuses on technique useful to a systems analyst doing requirements determination.

Mission – reason for existence
Goals – General statements
Objectives – Specific measurable statements
Tactics and needs – Actions to accomplish the objectives
Information systems – Support for user actions

The model presents five tiers starting with some internal or external stimuli (e.g. problems, opportunities or directives) representing the need or the desire for some type of change.

Most often successful use of the requirements model expects that the business has documented specific goal and mission statements, often in a document that is called an enterprise or business model. Briefly an enterprise or business model attempts to answer the question "Why do we exist". This type of question can be asked and answered by every organizational level – corporate, division, region, department, section and so on.

In the requirements model, **business objectives** are specific statements of how the organizational goals can be achieved. For example, "increase profit by 10% each year" and "reduce customer complaints 15% each year" could be examples of business objectives. The objectives are business directed, always measurable, usually stated in terms of time and/ or money, and in the spirit of total quality management (TQM) often do not have an ending point so that continuous improvement and excellence becomes an ongoing objective.

Business tactics – are specific actions that are taken to realize the business objectives. The business tactics may or may not specifically involve the information systems of the business.

Information system objectives – are the information system accomplishments, such as printing reports, displaying calculations such as total price etc.

VIDEO SALES/RENTAL STORE INFORMATION SYSTEMS PROJECT

Mission statement

To be the video source of choice by successfully providing a generous selection of home video products for sale or rental at competitive prices.

Goals

1. Increase market share and maintain profitability
2. Offer superior customer assistance and browsing environment

Business objectives

(What we want to accomplish for the business)

1. Decrease checkout time for customers by at least 50%
2. Improve membership management by 50%
3. Increase memberships by 75% each year for the next two years.
4. Improve inventory management by 60%
5. Purchase at least one new store each calendar year for the next three years and then begin acquiring several stores each year thereafter.

Business tactics

(How we plan to accomplish business objectives)

1.1. Revise the checkout methods for rentals and sales to be more efficient and effective.
2.1. Revise the membership management to be more efficient and effective
3.1. Implement a marketing strategy to increase membership
4.1. Revise inventory management to be more effective and efficient.
5.1. Replace/implement accounting and financial systems

Information systems objectives

General objectives:

A. Provide just-in-time-time (JIT) training.
B. The systems we implement must be friendly and easy to learn and use.
C. The systems we implement must give considerations to security issues.

Specific objectives

1.1.1. Provide an automated system to assist with customer sales/rental check-outs.

2.1.1. Provide and maintain an automated membership database

a. provide current (up to date) membership information on demand
b. capability to add, change, and delete (remove) membership information.

2.1.2. Provide membership information reports such as (but not limited to):

a. least used memberships.
b. most used memberships.
c. Delinquent memberships (both money owing and outstanding rentals).

4.1.1. Provide and maintain an inventory database for both sales and rental items.

a. provide current (up to date) inventory information on demand
b. capability to add, change, and delete (remove) inventory information (sales and rental).

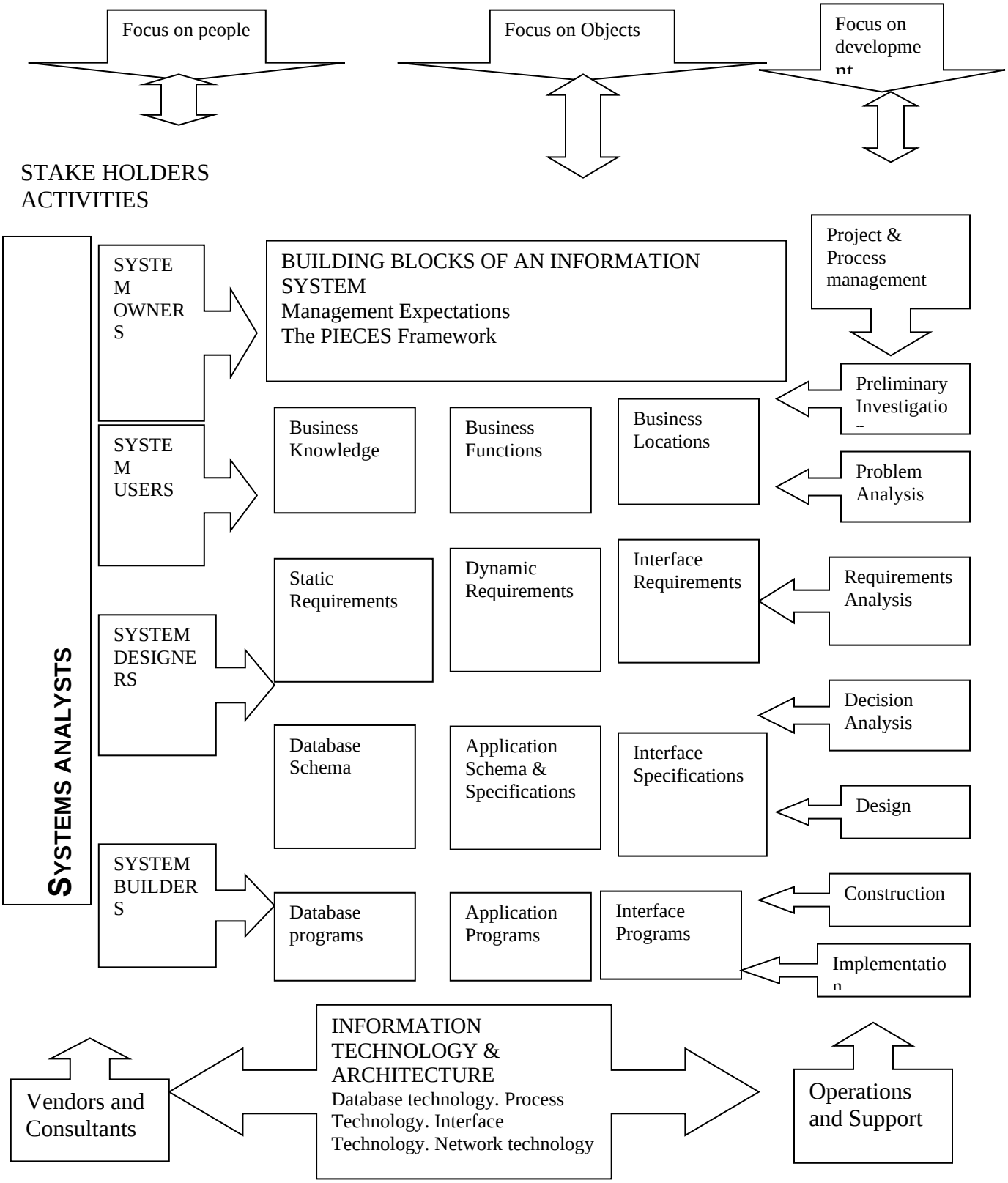
4.1.2. Provide inventory reports such as (but not limited to):

a. least popular rentals
b. most popular rentals
c. delinquent tape rentals outstanding
d. products "on order" (purchasing report) for sale and for rental items.

5.1.1. Provide sales reports such as (but not limited to):

a. sales for a time period (day, days, week, weeks, months, etc) by product code.
b. Rentals for a time period (same as above).

SYSTEMS ANALYSIS AND MODELING: An IS Development Approach



Recommended Reading

1. *Weatherbe, James C, (1988),* Readings in Information Systems: a managerial perspective, West Publishing Company, the United States of America.
2. *Jeffrey L. Whitten,* Systems Analysis and Design Methods, Higher Education Press, McGraw-Hill Companies, 2001.
3. Software Requirements, Second Edition by Karl E. Wiegers, 2003.
4. Software Engineering: A Practitioner's Approach, 4th Edition by Rogers S Pressman.