Thakur Educational Trust's (Regd.)

THAKUR RAMNARAYAN COLLEGE OF ARTS & COMMERCE

ISO 21001:2018 Certified

PROGRAMME: B.Sc (I.T)

CLASS: S.Y.B.Sc (I.T)

SUBJECT NAME: SOFTWARE ENGINEERING

SEMESTER: IV

FACULTY NAME: Ms. SMRITI DUBEY

UNIT II

Chapter 1 – Socio Technical System

Concepts:

Essential Characteristics of Socio – Technical System

Emergent System Properties

System Engineering

Components of System such as Organization, People and Computers

Legacy Systems

Socio Technical Systems

Software and hardware are interdependent. Without the hardware, a software is an abstraction. **When you put hardware and software together, you create a system**. This system will be able to carry out multiple complex computations and return the result to its environment. This illustrates one of the fundamental characteristics of the system. **Socio-technical system is basically a study of how any technology is used and produced.** This helps us to identify the ethical errors in technical and social aspects of the systems. **Socio-technical system is a mixture of people and technology.**

Socio-technical systems include:

People: People can be individuals or in groups. We also need to consider their roles and agencies. An organization employs the people, who build and make use of hardware and software, operate within law and regulations, and share and maintain the data.

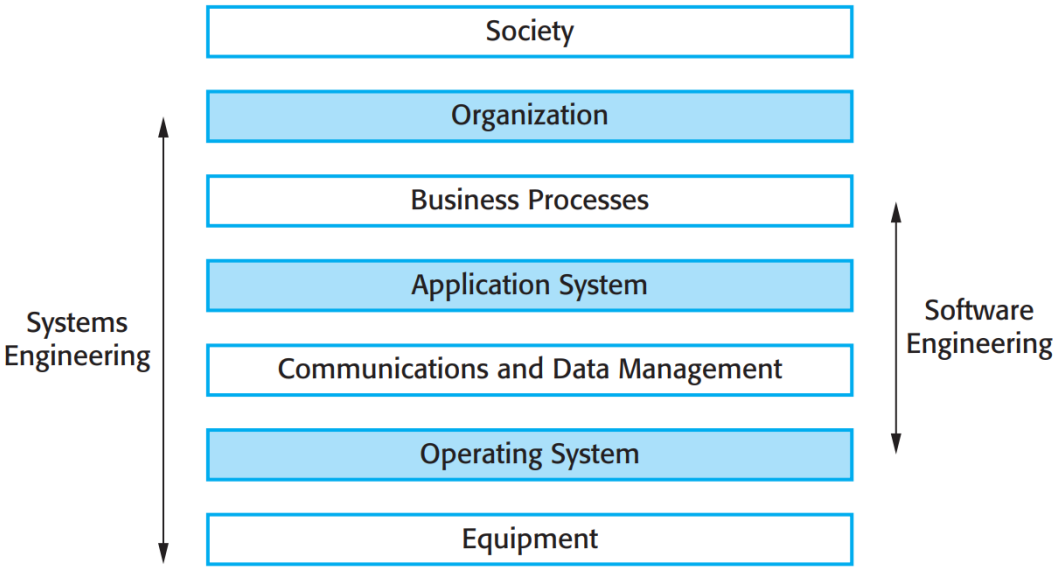
Hardware: The classical meaning if the technology is hardware. It involves mainframe, workstations, peripheral, connecting devices. There is no way for a socio-technical system to be without any kind of hardware component.

Software: Software is nothing but an executable code. Software includes operating system, utilities, application programs. Software is an integral part of the socio-technical system. Software often incorporates social rules and procedures as a part of the design, i.e., optimize these parameters, store the data in these formats, ask for these data, etc.

Law and regulations: There might be laws about the protection of privacy, or regulations of chips testing in military use, etc. Laws and regulations set by organization and government need to be followed. They carry special societal sanctions if the violators are caught.

Data: The design of the socio-technical systems design involve what data are collected, to whom the data should be available and in which formats the data should be stored.

Socio - technical systems are so complex that it is practically impossible to understand them as a whole. Rather, you have to view them as layers, as shown in Figure. These layers make up the sociotechnical systems stack:



1. **The equipment layer** - This layer is composed of hardware devices, some of which may be computers.

2. **The operating system layer** - This layer interacts with the hardware and provides a set of common facilities for higher software layers in the system.

3. **The communications and data management layer** - This layer extends the operating system facilities and provides an interface that allows interaction with more extensive functionality, such as access to remote systems, access to a system database, etc. This is sometimes called middleware, as it is in between the application and the operating system.

4. **The application layer** - This layer delivers the application-specific functionality that is required. There may be many different application programs in this layer.

5. **The business process layer** - At this level, the organizational business processes, which make use of the software system, are defined and enacted.

6. **The organizational layer** - This layer includes higher-level strategic processes as well as business rules, policies, and norms that should be followed when using the system.

7. **The social layer** - At this layer, the laws and regulations of society that govern the operation of the system is defined.

Essential Characteristics of Socio Technical Systems

Sociotechnical systems include one or more technical systems but, crucially, also include people who understand the purpose of the system within the system itself. Sociotechnical systems have defined operational processes and people (the operators) are inherent parts of the system. They are governed by organizational policies and rules and may be affected by external constraints such as national laws and regulatory policies. For example, this book was created through a sociotechnical publishing system that includes various processes and technical systems.

Sociotechnical systems have three characteristics that are particularly important when considering security and dependability:

1. They have emergent properties that are properties of the system as a whole, rather than associated with individual parts of the system. Emergent properties depend on both the system components and the relationships between them. Given this complexity, the emergent properties can only be evaluated once the system has been assembled. Security and dependability are emergent system properties.

2. **They are often nondeterministic** - This means that when presented with a specific input, they may not always produce the same output. The system's behavior depends on the human operators and people do not always react in the same way. Furthermore, use of the system may create new relationships between the system components and hence change its emergent behavior. System faults and failures may therefore be transient, and people may disagree about whether or not a failure has actually occurred.

3. **The extent to which the system supports organizational objectives does not just depend on the system itself**. It also depends on the stability of these objectives, the relationships, and conflicts between organizational objectives and how people in the organization interpret these objectives. New management may reinterpret the organizational objectives that system was designed to support so that a 'successful' system may then be seen as a 'failure'.

Emergent System Properties

The complex relationships between the components in a system mean that a system is more than simply the sum of its parts. It has properties that are properties of the system as a whole. These 'emergent properties' cannot be attributed to any specific part of the system. Rather, they only emerge once the system components have been integrated. Some of these properties, such as weight, can be derived directly from the comparable properties of subsystems. More often, however, they result from complex subsystem interrelationships.

There are two types of emergent properties:

1. **Functional emergent properties** when the purpose of a system only emerges after its components are integrated. For example, a bicycle has the functional property of being a transportation device once it has been assembled from its components.

2. **Non-functional emergent properties**, which relate to the behavior of the system in its operational environment. Reliability, performance, safety, and security are examples of emergent properties. These are critical for computer-based systems, as failure to achieve a minimum defined level in these properties usually makes the system unusable. Some users may not need some of the system functions, so the system may be acceptable without them. However, a system

that is unreliable or too slow is likely to be rejected by all its users.

Property	Description
Volume	The volume of a system (the total space occupied) varies depending on how the component assemblies are arranged and connected.
Reliability	System reliability depends on component reliability but unexpected interactions can cause new types of failures and therefore affect the reliability of the system.
Security	The security of the system (its ability to resist attack) is a complex property that cannot be easily measured. Attacks may be devised that were not anticipated by the system designers and so may defeat built-in safeguards.
Repairability	This property reflects how easy it is to fix a problem with the system once it has been discovered. It depends on being able to diagnose the problem, access the components that are faulty, and modify or replace these components.
Usability	This property reflects how easy it is to use the system. It depends on the technical system components, its operators, and its operating environment.

Figure – Examples of Emergent Properties

System reliability is a good example of an emergent property. Because of component inter-dependencies, faults can be propagated through the system. System failures often occur because of unforeseen inter-relationships between components. It is practically impossible to anticipate all possible component relationships. Software reliability measures may give a false picture of the overall system reliability.

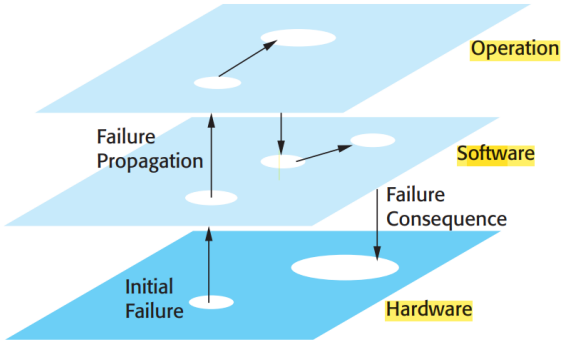
System reliability is influenced by:

Hardware reliability: What is the probability of a hardware component failing and how long does it take to repair that component?

Software reliability: How likely is it that a software component will produce an incorrect output. Software failure is usually distinct from hardware failure in that software does not wear out.

Operator reliability: How likely is it that the operator of a system will make an error?

Failure propagation



cause of failure - hardware, software and operational failure

Hardware, software, and operator reliability are not independent. Figure above shows how failures at one level can be propagated to other levels in the system.Hardware failure can generate spurious signals that are outside the range of inputsexpected by the software. The software can then behave unpredictably and produce unexpected outputs. These may confuse and consequently stress the system operator.

Operator error is most likely when the operator is feeling stressed. So a hardware failure may then mean that the system operator makes mistakes which, in turn, couldlead to further software problems or additional processing. This could overload the hardware, causing more failures and so on. Thus, the initial failure, which might be recoverable, can rapidly develop into a serious problem that may result in a complete shutdown of the system.

The reliability of a system depends on the context in which that system is used. However, the system's environment cannot be completely specified, nor can the system designers place restrictions on that environment for operational systems. Different systems operating within an environment may react to problems in unpredictable ways, thus affecting the reliability of all of these systems.

Systems engineering

Systems engineering encompasses all of the activities involved in procuring, specifying, designing, implementing, validating, deploying, operating, and maintaining sociotechnical systems. Systems engineers are not just concerned with software but also with hardware and the system's interactions with users and its environment. They must think about the services that the system provides, the constraints under which the system must be built and operated, and the ways in which the system is used to fulfill its purpose or purposes.

There are three overlapping stages (Figure) in the lifetime of large and complex sociotechnical systems:

1**. Procurement or acquisition** - During this stage, the purpose of a system is decided; high-level system requirements are established; decisions are made on how functionality will be distributed across hardware, software, and people; and the components that will make up the system are purchased.

2. **Development -** During this stage, the system is developed. Development processes include all of the activities involved in system development such as requirements definition, system design, hardware and software engineering, system integration, and testing. Operational processes are defined and the training courses for system users are designed.

3. **Operation -** At this stage, the system is deployed, users are trained, and the system is brought into use. The planned operational processes usually then have to change to reflect the real working environment where the system is used. Overtime, the system evolves as new requirements are identified. Eventually, the system declines in value and it is decommissioned and replaced.

These stages are not independent. Once the system is operational, new equipment and software may have to be procured to replace obsolete system components, to provide new functionality, or to cope with increased demand. Similarly, requests for changes during operation require further system development.

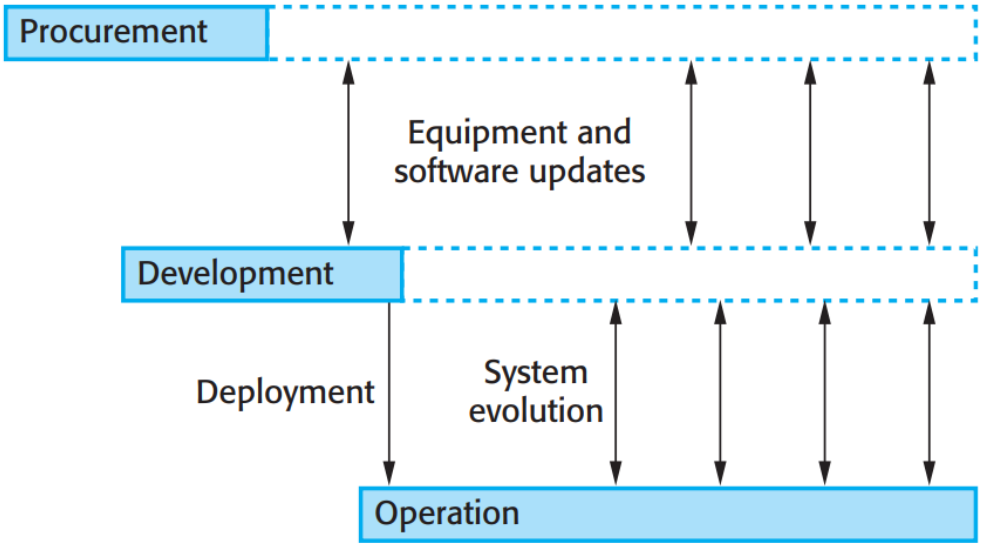


Figure – Stages of System Engineering

The overall security and dependability of a system is influenced by activities at all of these stages. Design options may be restricted by procurement decisions on the scope of the system and on its hardware and software. It may be impossible to implement some kinds of system safeguards. They may introduce vulnerabilities that could lead to future system failures. Human errors made during the specification, design, and development stages may mean that faults are introduced into the system. Inadequate testing may mean that faults are not discovered before a system is deployed. During operation, errors in configuring the system for deployment may lead to further vulnerabilities. System operators may make mistakes in using the system. Assumptions made during the original procurement may be forgotten when system changes are made and, again, vulnerabilities can be introduced into the system.

Components of System such as Organization, People and Computers

Sociotechnical systems are enterprise systems that are intended to help deliver a business goal. This might be to increase sales, reduce material used in manufacturing, collect taxes, maintain a safe airspace, etc. Because they are embedded in an organizational environment, the procurement, development, and use of these systems are influenced by the organization's policies and procedures, and by its working culture. **The users of the system are people who are influenced by the way the organization is managed and by their interactions with other people inside and outside of the organization.**

Organizational factors from the system's environment that may affect the requirements, design, and operation of a sociotechnical system include:

1. **Process changes** - The system may require changes to the work processes in the environment. If so, training will certainly be required. If changes are significant, or if they involve people losing their jobs, there is a danger that the users will resist the introduction of the system.

2. **Job changes** - New systems may de-skill the users in an environment or cause them to change the way they work. If so, users may actively resist the introduction of the system into the organization. Designs that involve managers having to change their way of working to fit a new computer system are often resented. The managers may feel that their status in the organization is being reduced by the system.

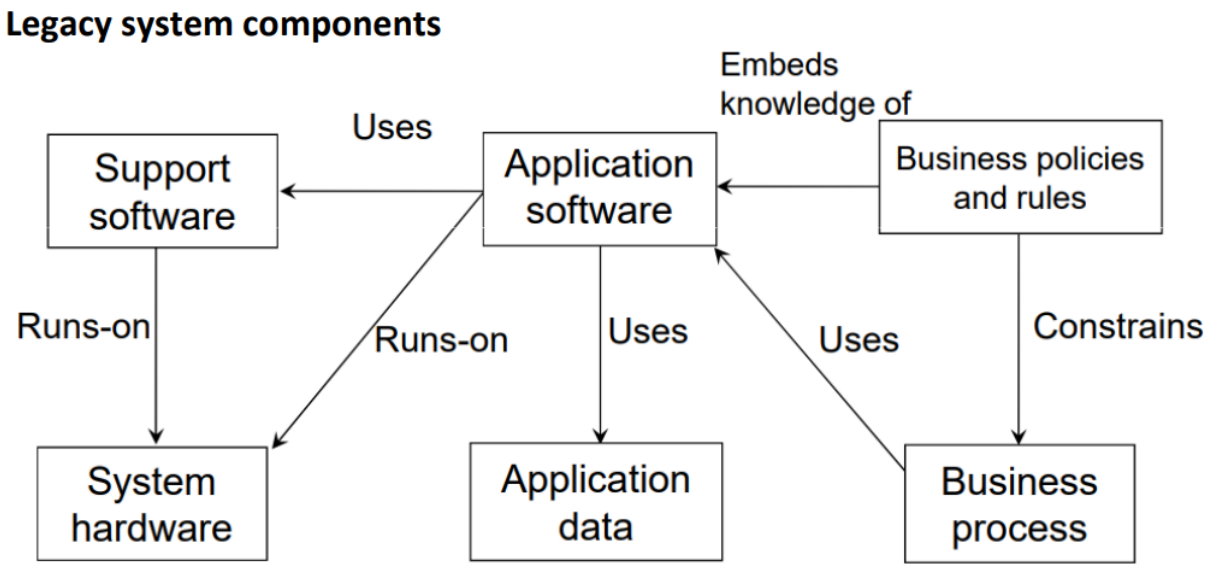
3. **Organizational changes** - The system may change the political power structure in an organization. For example, if an organization is dependent on a complex system, those who control access to that system have a great deal of political power.

When you are trying to develop sociotechnical systems, you need to understand the organizational environment in which they are used. If you don't, the systems may not meet business needs and users and their managers may reject the system.

Legacy Systems

Legacy systems are sociotechnical computer-based systems that have been developed in the past, often using older or obsolete technology. These systems include not only hardware and software but also legacy processes and procedures—old ways of doing things that are difficult to change because they rely on legacy software. Changes to one part of the system inevitably involve changes to other components.

Legacy systems are often business-critical systems. They are maintained because it is too risky to replace them.



Legacy system components

• **System Hardware** – In many cases legacy systems have been written for mainframe hardware that is no longer available that is expensive to maintain and that may not be compatible with current organizational IT purchasing policies.

• **Support software** – The legacy system may rely on range of support software from the operating system and utilities provided by the hardware manufacturer through to the compilers used for system development.

• **Application software** – The application system that provides the business services is usually composed of a number of separate programs that have been developed at different times.

• **Application data** – These are the data that are processed by the application system. In many legacy systems, an immense volume of data has accumulated over the lifetime of the system. This data may be inconsistent and may be duplicated in several files.

• **Business processes** – These are processes that are used in the business to achieve some business objective. An example of a business process in an insurance company would be issuing an insurance policy; in a manufacturing company a business process would be accepting all order for products and setting up the associated manufacturing process.

• **Business policies and rules** – These are definitions of how the business should be carried out and constraints on the business. Use of the legacy application system may be embedded in these policies and rules.