Workbook

Information Systems Engineering
(SE – 483)

Name _____

Roll No _____

Batch _____

Year _____

Department _____

Workbook

Information Systems Engineering
(SE – 483)

Prepared by

Dr. Sh. M. Wahabuddin Usmani

Approved by

Chairman
Department of Computer Science & Information Technology

Table of Contents

PRACTICAL No: 01

Business Process / Product Engineering

Software engineering occurs as a consequence of a process called *system engineering.* Instead of concentrating solely on software, system engineering focuses on a variety of elements, analyzing, designing, and organizing those elements into a system that can be a product, a service, or a technology for the transformation of information or control.

The system engineering process is called ***business process engineering*** when the context of the engineering work focuses on a business enterprise. When a product (in this context, a product includes everything from a wireless telephone to an air traffic control system) is to be built, the process is called ***product engineering***.

Both business process engineering and product engineering attempt to bring order to the development of computer-based systems. Although each is applied in a different application domain, both strive to put software into context.

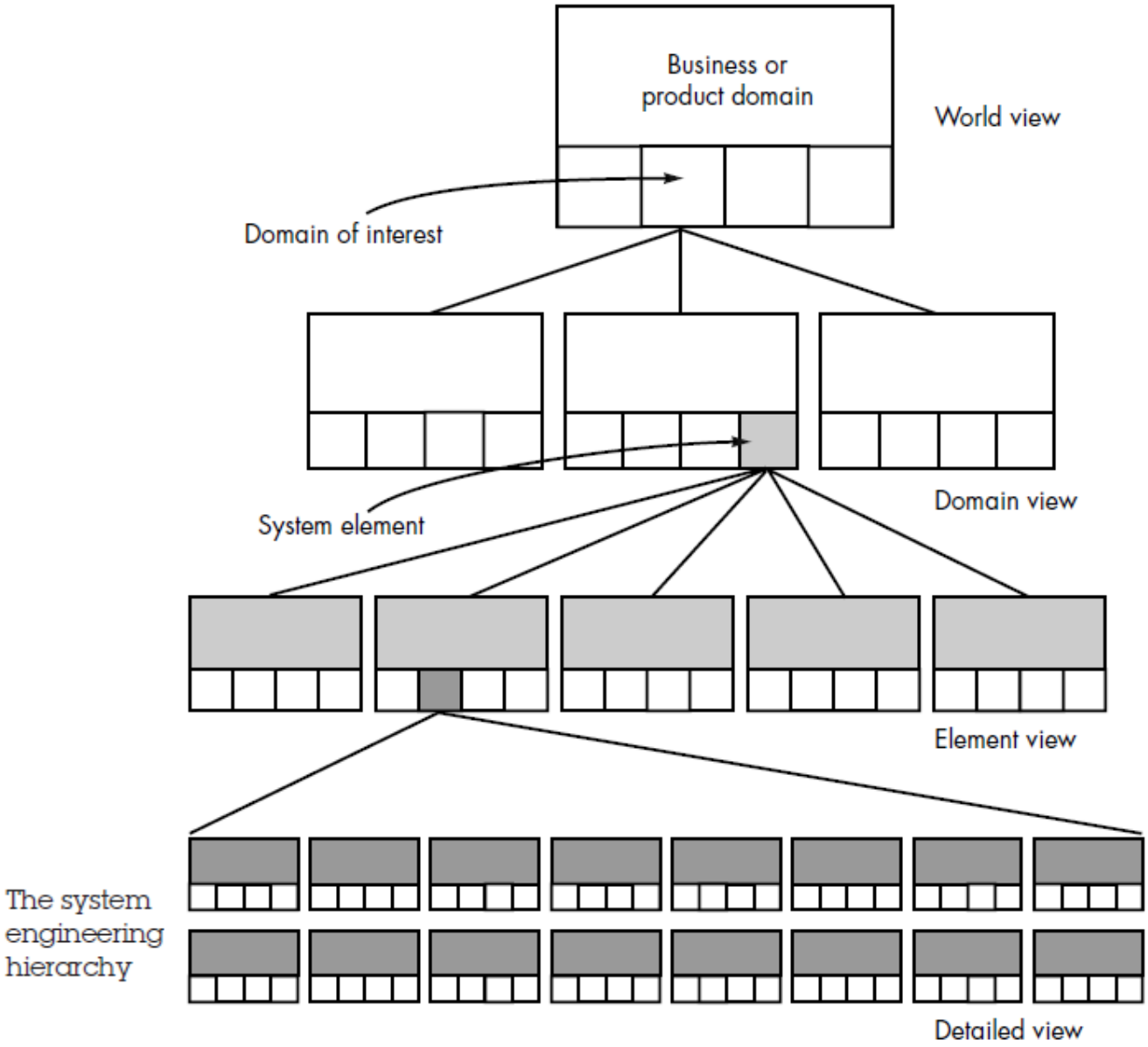
That is, both business process engineering and product engineering work to allocate a role for computer software and, at the same time, to establish the links that tie software to other elements of a computer-based system.

Required:

With reference to your final year project, explain in which area, Business Process/Product engineering it falls and why?

PRACTICAL No: 02

System Engineering Hierarchy



Required:

Draw the system engineering hierarchy diagram for your project

PRACTICAL No: 03
System Modeling

To construct a system model, the engineer should consider a number of restraining factors:

1. *Assumptions* that reduce the number of possible permutations and variations, thus enabling a model to reflect the problem in a reasonable manner. As an example, consider a three-dimensional rendering product used by the entertainment industry to create realistic animation. One domain of the product enables the representation of 3D human forms. Input to this domain encompasses the ability to specify movement from a live human actor, from video, or by the creation of graphical models. The system engineer makes certain assumptions about the range of allowable human movement (e.g., legs cannot be wrapped around the torso) so that the range of inputs and processing can be limited.

2. *Simplifications* that enable the model to be created in a timely manner. To illustrate, consider an office products company that sells and services a broad range of copiers, faxes, and related equipment. The system engineer is modeling the needs of the service organization and is working to understand the flow of information that spawns a service order. Although a service order can be derived from many origins, the engineer categorizes only two sources: internal demand and external request. This enables a simplified partitioning of input that is required to generate the service order.

3. *Limitations* that help to bound the system. For example, an aircraft avionics system is being modeled for a next generation aircraft. Since the aircraft will be a two-engine design, the monitoring domain for propulsion will be modeled to accommodate a maximum of two engines and associated redundant systems.

4. *Constraints* that will guide the manner in which the model is created and the approach taken when the model is implemented. For example, the technology infrastructure for the three-dimensional rendering system described previously is a single G4-based processor. The computational complexity of problems must be constrained to fit within the processing bounds imposed by the processor.

5. *Preferences* that indicate the preferred architecture for all data, functions, and technology. The preferred solution sometimes comes into conflict with other restraining factors. Yet, customer satisfaction is often predicated on the degree to which the preferred approach is realized. The resultant system model (at any view) may call for a completely automated solution, a semi-automated solution, or a non-automated approach. In fact, it is often possible to characterize models of each type that serve as alternative solutions to the problem at hand. In essence, the system engineer simply modifies the relative influence of different system elements (people, hardware, software) to derive models of each type.

Required:

Describe the restraining factors mentioned above for your final year project

PRACTICAL No: 04
Goals of System Modeling – Data Architecture

The goal of business process engineering (BPE) is to define architectures that will enable a business to use information effectively.

However, the price for this change is largely borne by the IT [information technology] organizations that must support this polyglot configuration. Today, each IT organization must become, in effect, its own systems integrator and architect. It must design, implement, and support its own unique configuration of heterogeneous computing resources, distributed logically and geographically throughout the enterprise, and connected by an appropriate enterprise-wide networking scheme.

Moreover, this configuration can be expected to change continuously, but unevenly, across the enterprise, due to changes in business requirements and in computing technology. These diverse and incremental changes must be coordinated across a distributed environment consisting of hardware and software supplied by dozens, if not hundreds, of vendors. And, of course, we expect these changes to be seamlessly incorporated without disrupting normal operations and to scale gracefully as those operations expand.

When taking a world view of a company's information technology needs, there is little doubt that system engineering is required. Not only is the specification of the appropriate computing architecture required, but the software architecture that populates the "unique configuration of heterogeneous computing resources" must be developed. Business process engineering is one approach for creating an overall plan for implementing the computing architecture.

Three different architectures must be analyzed and designed within the context of business objectives and goals:

- data architecture
- applications architecture
- technology infrastructure

The *data architecture* provides a framework for the information needs of a business or business function. The individual building blocks of the architecture are the data objects that are used by the business. A data object contains a set of attributes that define some aspect, quality, characteristic, or descriptor of the data that are being described. For example, an information engineer might define the data object **customer.** To more fully describe **customer,** the following attributes are defined:

Object: Customer

Attributes:

name
company name
job classification and purchase authority
business address and contact information
product interest(s)
past purchase(s)
date of last contact
status of contact

Once a set of data objects is defined, their relationships are identified. A *relationship* indicates how objects are connected to one another. As an example, consider the objects: **customer,** and **product A.** The two objects can be connected by the relationship *purchases;* that is, a customer purchases product A or product A is purchased by a customer. The data objects (there may be hundreds or even thousands for a major business activity) flow between business functions, are organized within a database, and are transformed to provide information that serves the needs of the business.

Required:

For your project as described above construct Data Architecture.

PRACTICAL No: 05
Goals of System Modeling – Application Architecture

The *application architecture* encompasses those elements of a system that transform objects within the data architecture for some business purpose. In the context of this book, we consider the application architecture to be the system of programs (software) that performs this transformation. However, in a broader context, the application architecture might incorporate the role of people (who are information transformers and users) and business procedures that have not been automated.

Required:

For your project describe the Application Architecture.

PRACTICAL No: 06
Goals of System Modeling – Technology Infrastructure

The *technology infrastructure* provides the foundation for the data and application architectures. The infrastructure encompasses the hardware and software that are used to support the application and data. This includes computers, operating systems, networks, telecommunication links, storage technologies, and the architecture (e.g., client/server) that has been designed to implement these technologies

Required:

Describe Technology Infrastructure for your project.

PRACTICAL No: 07
Business Process Engineering Hierarchy

To model the system architectures described earlier, a hierarchy of business process engineering activities is defined. Referring to Figure, the world view is achieved through *information strategy planning* (ISP). ISP views the entire business as an entity and isolates the domains of the business (e.g., engineering, manufacturing, marketing, finance, sales) that are important to the overall enterprise. ISP defines the data objects that are visible at the enterprise level, their relationships, and how they flow between the business domains.

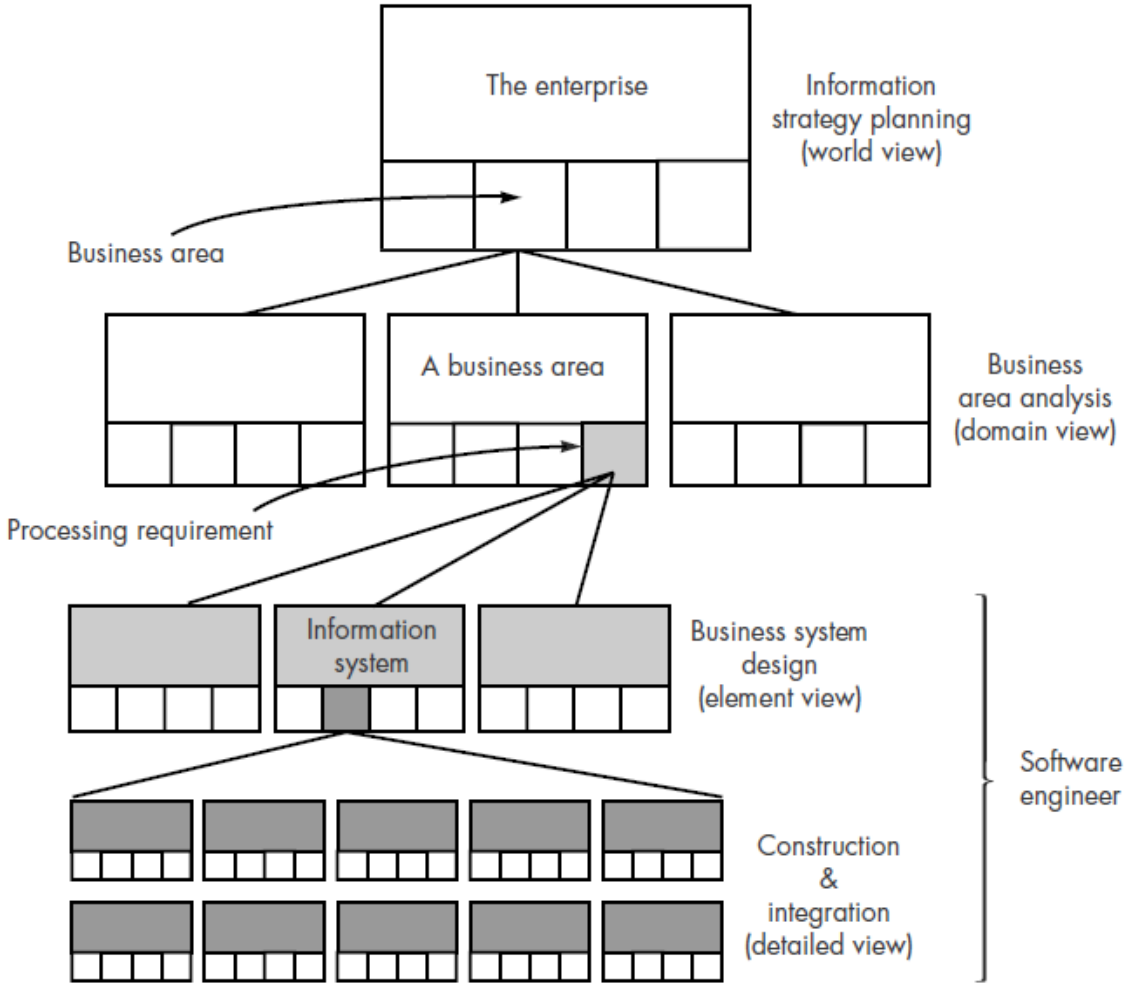
The domain view is addressed with a BPE activity called *business area analysis* (BAA). Hares describes BAA in the following manner:

BAA is concerned with identifying in detail data (in the form of entity [data object] types) and function requirements (in the form of processes) of selected business areas [domains] identified during ISP and ascertaining their interactions (in the form of matrices). It is only concerned with specifying what is required in a business area.

As the system engineer begins BAA, the focus narrows to a specific business domain. BAA views the business area as an entity and isolates the business functions and procedures that enable the business area to meet its objectives and goals. BAA, like ISP, defines data objects, their relationships, and how data flow. But at this level, these characteristics are all bounded by the business area being analyzed. The outcome of BAA is to isolate areas of opportunity in which information systems may support the business area.

Once an information system has been isolated for further development, BPE makes a transition into software engineering. By invoking a *business system design* (BSD) step, the basic requirements of a specific information system are modeled and these requirements are translated into data architecture, applications architecture, and technology infrastructure.

The final BPE step—*construction and integration* focuses on implementation detail. The architecture and infrastructure are implemented by constructing an appropriate database and internal data structures, by building applications using software components, and by selecting appropriate elements of a technology infrastructure to support the design created during BSD. Each of these system components must then be integrated to form a complete information system or application. The integration activity also places the new information system into the business area context, performing all user training and logistics support to achieve a smooth transition.



The business process engineering hierarchy

Required:

Construct the Business Hierarchy Diagram for your project.

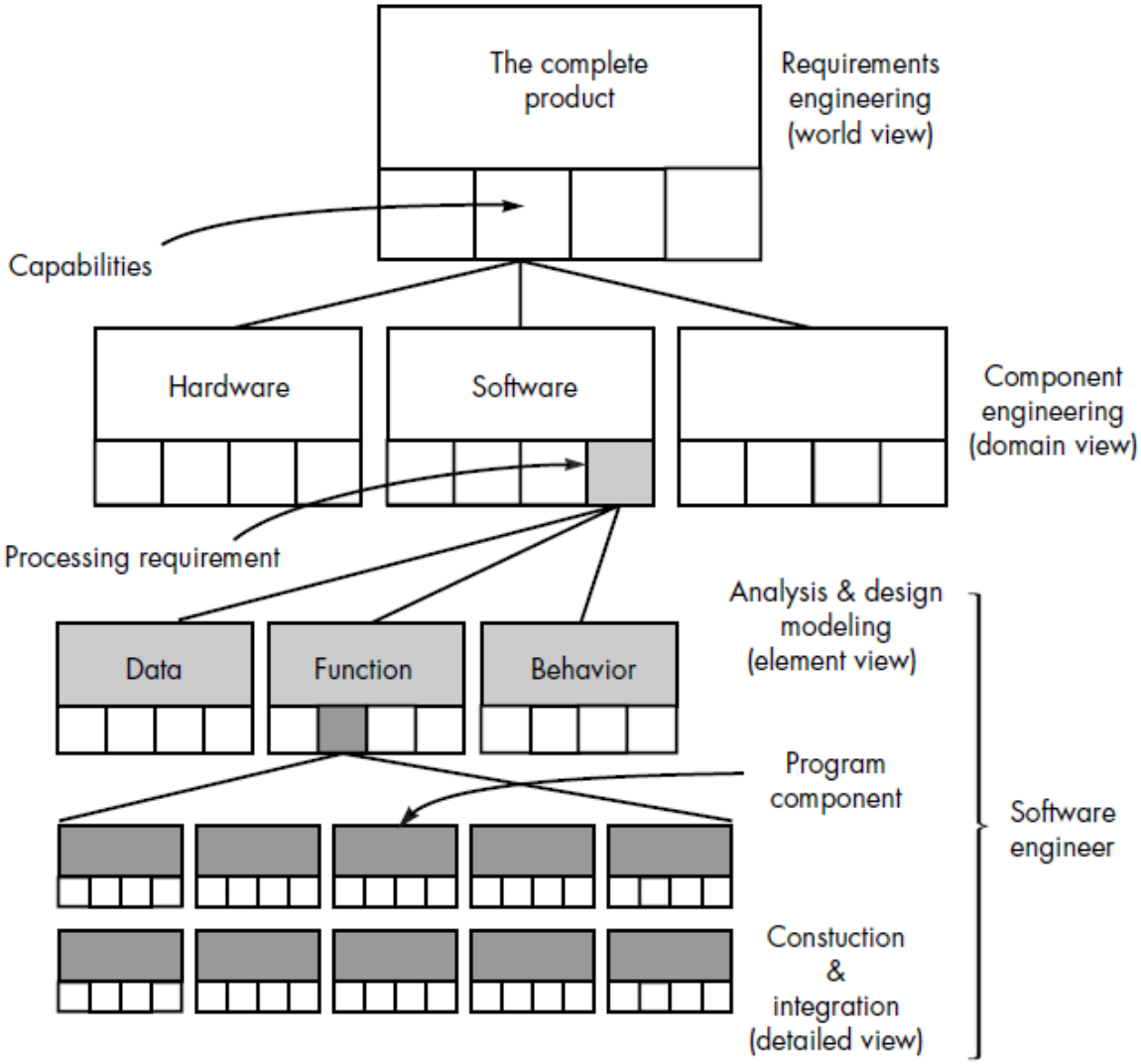
PRACTICAL No: 08
Product Engineering Hierarchy

The goal of product engineering is to translate the customer's desire for a set of defined capabilities into a working product. To achieve this goal, product engineering—like business process engineering—must derive architecture and infrastructure. The architecture encompasses four distinct system components: software, hardware, data (and databases), and people. A support infrastructure is established and includes the technology required to tie the components together and the information (e.g., documents, CD-ROM, video) that is used to support the components.

Referring to Figure, the world view is achieved through *requirements engineering.* The overall requirements of the product are elicited from the customer. These requirements encompass information and control needs, product function and behavior, overall product performance, design and interfacing constraints, and other special needs. Once these requirements are known, the job of requirements engineering is to allocate function and behavior to each of the four components noted earlier.

Once allocation has occurred, *system component engineering* commences. System component engineering is actually a set of concurrent activities that address each of the system components separately: software engineering, hardware engineering, human engineering, and database engineering. Each of these engineering disciplines takes a domain-specific view, but it is important to note that the engineering disciplines must establish and maintain active communication with one another. Part of the role of requirements engineering is to establish the interfacing mechanisms that will enable this to happen.

The element view for product engineering is the engineering discipline itself applied to the allocated component. For software engineering, this means *analysis and design modeling activities* (covered in detail in later chapters) and *construction and integration activities* that encompass code generation, testing, and support steps. The analysis step models allocated requirements into representations of data, function, and behavior. Design maps the analysis model into data, architectural, interface, and software component-level designs.



The product engineering hierarchy

Required:

Construct Product Engineering Hierarchy Diagram for your project.