# System Integration & Architecture

ROJO, J/ALIWATE, D/EDO, A EDO/UY, ULRICH/ODARVE, C

# Introduction

- Many systems are built to easy, improve and transform organizations.
- Some organizations have many departments which run systems which are independent of each other.
- And systems built sometimes, may not have an abstract view (**architecture**) which leads to failure of system interoperability.
- There is need to have architectural view of the system as a priority to help in the design to avoid the likeliness of system failure.

# Introduction

- Besides after the system has been designed and developed in consideration of the **size of the organization**, i.e. most especially when the organization is large, need is required to integrate such systems to ensure **flexibility**, **Speed**, **Cost** , **Standardization**, **Data integrity**, **reliability** and **robustness**.
- This can help Information Technology (IT), energy, and financial services industry among others to have an easy to use integrated system.

# What is System Integration

- Systems Integration (SI) is the process of ensuring that a proposed system works seamlessly with other existing or planned systems. It involves applying specific approaches, tools, and techniques to combine different system components effectively. Successful SI depends on understanding the critical success factors and following best practices. The process often uses system architectures as a foundation, allowing the evaluation of how well these architectures provide the necessary capabilities to solve integration challenges.The course focuses on how a proposed system will be integrated with other existing or planned systems.

# What is System Integration

- It addresses the Integration problem using architectures as the basis and then addresses the evaluation of the architectures in terms of the capabilities they provide.

# System Integration Applications

1. **Business Process Integration** – Aligning and connecting different business processes to work seamlessly across the organization.
2. **Legacy System Integration** – Connecting older systems with new technologies to maintain functionality and data consistency.
3. **New System Integration** – Ensuring newly developed systems operate smoothly with existing systems.
4. **Business-to-Business (B2B) Integration** – Linking systems between different organizations for efficient data exchange and collaboration.

# System Integration Applications

5. **COTS Product Integration** – Integrating commercial-off-the-shelf software products with other systems.
6. **Interface Control and Management** – Managing and coordinating interfaces between system components.
7. **Testing** – Verifying that integrated systems function correctly and meet requirements.
8. **Integrated Program Management** – Coordinating multiple projects or programs to achieve system integration objectives.
9. **Integrated Business Continuity Planning (BCP)** – Ensuring that integrated systems support business continuity in case of disruptions.

# Aims

- Assess emerging system integration tools and methodologies to be used in solving specific integration problems.
- To provide the students an understanding of the technical and business process issues involved in systems integration.

# Indicative content

- The System of Systems Integration Problem
  - Human, Organizational, Societal Cultural, Economic, and Technological aspects;
    - Processes, approaches, drivers, tools and techniques required for successful SI, critical success factors, and best practices in Systems Integration;
    - The Role of Architectures in Systems Integration;
    - Integration in a System of Systems and a Federation 60 of Systems;
    - Model Based Architecture, Design, and Integration;
    - Systems of Systems Interoperability;
    - Evaluation of architectures;
    - Measures of Performance and Effectiveness;

# Indicative content

- Assessment of System Capabilities;
- Analysis of Alternatives;
- Case studies and examples from the Information Technology (IT), energy, and financial services industry to illustrate the concepts discussed.

- The theory and practice of business process integration, legacy integration, new systems integration, business-to-business integration, integration of commercial-off-the-shelf (COTS) products, interface control and management, testing, integrated program management, integrated Business Continuity Planning (BCP). Specific focus will be given to issues of interface integration and interoperability of systems.

# Key terminologies in this course

- Various key terminologies shall be used throughout this course as follows
- System
- Systems thinking
- System Integration
- System Architecture
- Project

# System

- An array of components designed to accomplish a particular objective according to plan. Many sub-systems many be designed which later on are combined together to form a system which is intended  to achieve a specific objective which may be set by the Project manager.

# Systems thinking

- Is a way of understanding an entity in terms of its purpose, as three steps
- The three major steps followed in systems thinking

1. Identify a containing whole (system), of which the thing to be explained is a part.
2. Explain the behavior or properties of the containing whole.
3. Explain the behavior or properties of the thing to be explained in terms of its *role(s)or function(s) within its containing whole*

*(Ackoff, 1981)*

# System Integration

- Is the combination of inter-related elements to achieve a common objective (s).

# System Architecture

- The architecture of a system defines its high-level structure, exposing its gross organization as a collection of interacting components.
- Elements needed to model a software architecture include:
  - Components, Connectors, Systems, Properties and Styles.

# What is a project?

- From the key terms described above, a system developer and architects cannot do anything without first establishing various projects. These projects may be new or existing.
- So it is inevitable to first understand what a project is, factors that influence the project, who the owners are and many more as discussed below.

# What Is a Project?

- A project is a temporary endeavor undertaken to accomplish a unique product or service
- Attributes of projects
  - unique purpose
  - temporary
  - require resources, often from various areas
  - should have a primary sponsor and/or customer
  - involve uncertainty

# Where do information Systems Projects Originate (Sources of Projects)?

New or changed IS development projects come from **problems, opportunities**, and **directives** and are always subject to one or more *constraints*.

1. **Problems** – may either be current, suspected, or anticipated. Problems are undesirable situations that prevent the business from fully achieving its purpose, goals, and objectives (users discovering real problems with existing IS).

2. An **Opportunity** – is a chance to improve the business even in the absence of specific problems. This means that the business is hoping to create a system that will help it with increasing its revenue, profit, or services, or decreasing its costs.

3. **A Directive** – is a new requirement that is imposed by management, government, or some external influence i.e. are mandates that come from either an internal or external source of the business.

# Projects Cannot Be Run in Isolation

- Projects must operate in a broad organizational environment
- Project managers need to take a holistic or systems view of a project and understand how it is situated within the larger organization

# Stakeholders

- Stakeholders are the people involved in or affected by project activities
- Stakeholders include
  - the project sponsor and project team
  - support staff
  - customers
  - users
  - suppliers
  - opponents to the project

# Importance of Stakeholders

- Project managers must take time to identify, understand, and manage relationships with all project stakeholders
- Using the four frames of organizations can help meet stakeholder needs and expectations
- Senior executives are very important stakeholders

# What Helps Projects Succeed?

According to the Standish Group's report "CHAOS 2001: A Recipe for Success," the following items help IT projects succeed, in order of importance:

- **Executive support**
- User involvement
- Experienced project manager
- Clear business objectives
- Minimized scope
- Standard software infrastructure
- Firm basic requirements
- Formal methodology
- Reliable estimates

# Understanding Organizations

## We can analyze a formal organization using the following 4 (four) fram

| | **Structural frame:** | **Human resources frame:** |
|---|---|---|
| | Focuses on roles and responsibilities, coordination and control. Organizational charts help define this frame. | Focuses on providing harmony between needs of the organization and needs of people. |
| | **Political frame:** | **Symbolic frame:** |
| | Assumes organizations are coalitions composed of varied individuals and interest groups. Conflict and power are key issues. | Focuses on symbols and meanings related to events. Culture is important. |

# Many Organizations Focus on the Structu Frame

- Most people understand what organizational charts are
- Many new managers try to change organizational structure when other changes are needed
- 3 basic organizational structures
  - Functional-
  - project
  - matrix

# Basic Organizational Structures

- Organizational structure depends on the company and/or the project.
- The structure helps define the roles and responsibilities of the members of the department, work group, or organization.
- It is generally a system of tasks and reporting policies in place to give members of the group a direction when completing projects.
- A good organizational structure will allow people and groups to work effectively together while developing hard work ethics and attitudes.
- The four general types of organizational structure are functional, divisional, matrix and project-based.

# Basic Organizational Structures

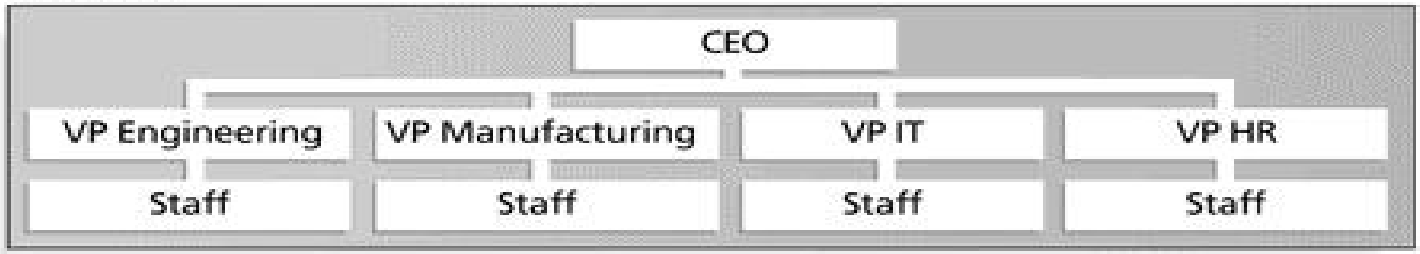
- **Functional Structure** - People who do similar tasks, have similar skills and/or jobs in an organization are grouped into a functional structure. The advantages of this kind of structure include quick decision making because the group members are able to communicate easily with each other. People in functional structures can learn from each other easier because they already possess similar skill sets and interests.
- **Divisional Structure -** In a divisional structure, the company will coordinate inter-group relationships to create a work team that can readily meet the needs of a certain customer or group of customers. The division of labor in this kind of structure will ensure greater output of varieties of similar products. An example of a divisional structure is geographical, where divisions are set up in regions to work with each other to produce similar products that meet the needs of the individual regions.

# Basic Organizational Structures

- **Matrix Structure** - Matrix structures are more complex in that they group people in two different ways: by the function they perform and by the product team they are working with. In a matrix structure the team members are given more autonomy and expected to take more responsibility for their work. This increases the productivity of the team, fosters greater innovation and creativity, and allows managers to cooperatively solve decision-making problems through group interaction.
- **Project Organization Structure -** In a project-organizational structure, the teams are put together based on the number of members needed to produce the product or complete the project. The number of significantly different kinds of tasks are taken into account when structuring a project in this manner, assuring that the right members are chosen to participate in the project.

# Basic Organizational Structure

Functional




Project


Matrix

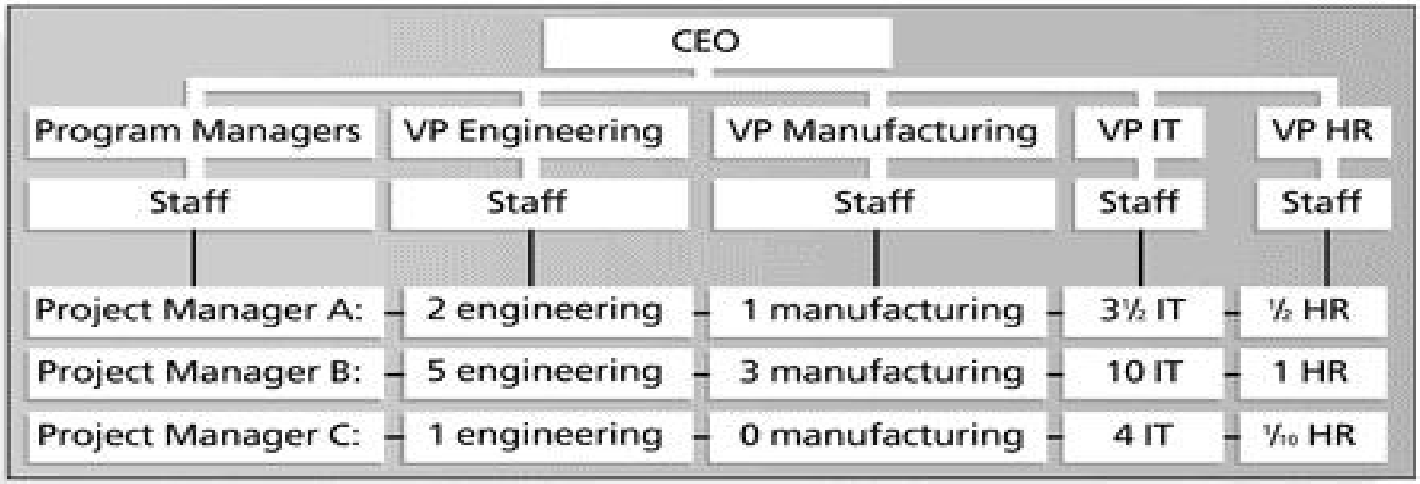
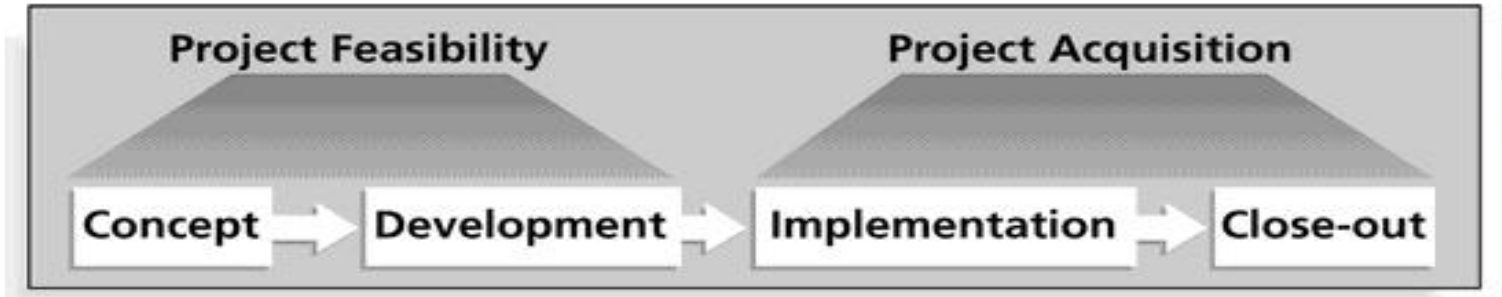



*Figure 2-2. Functional, Project, and Matrix Organizational Structures*

# Project Phases and the Project Life Cycle

- A project life cycle is a collection of project phases
- Project phases vary by project or industry, but some general phases include
  - concept
  - development
  - implementation
  - support

# Phases of the Project Life Cycle




| Sample deliverables for each phase | | | | |
|---|---|---|---|---|
| | Management plan | Project plan | Last work package | Completed work |
| | Preliminary cost estimate | Budgetary cost estimate | Definitive cost estimate | Lessons learned |
| | 3-level WBS | 6+-level WBS | Performance reports | Customer acceptance |

*Figure 2-3. Phases of the Project Life Cycle*

# Product Life Cycles

- Products also have life cycles
- **The Systems Development Life Cycle (SDLC)** is a framework for describing the phases involved in developing and maintaining information systems
- **Systems development projects can follow**
  - **Predictive models:** The scope of the project can be clearly articulated and the schedule and cost can be predicted.
  - **Adaptive models:** Projects are mission driven and component based, using time-based cycles to meet target dates.

# Predictive Life Cycle Models

- The waterfall model has well-defined, linear stages of systems development and support.
- The spiral model shows that software is developed using an iterative or spiral approach rather than a linear approach.
- The incremental release model provides for progressive development of operational software.
- The prototyping model is used for developing prototypes to clarify user requirements.
- The RAD model is used to produce systems quickly without sacrificing quality.

# Adaptive Life Cycle Models

- **Extreme Programming (XP):** Developers program in pairs and must write the tests for their own code. XP teams include developers, managers, and users.

- **Scrum:** Repetitions of iterative development are referred to as sprints, which normally last thirty days. Teams often meet every day for a short meeting, called a scrum, to decide what to accomplish that day. Works best for object-oriented technology projects and requires strong leadership to coordinate the work

# Distinguishing Project Life Cycles and Product Life Cycles

- The project life cycle applies to all projects, regardless of the products being produced
- Product life cycle models vary considerably based on the nature of the product
- Most large IT systems are developed as a series of projects
- Project management is done in all of the product life cycle phases

# Why Have Project Phases and Manageme Reviews?

- A project should successfully pass through each of the project phases in order to continue on to the next

- Management reviews (also called phase exits or kill points) should occur after each phase to **evaluate the project's progress**, **likely success**, and **continued compatibility** with organizational goals

# System Development Life Cycle

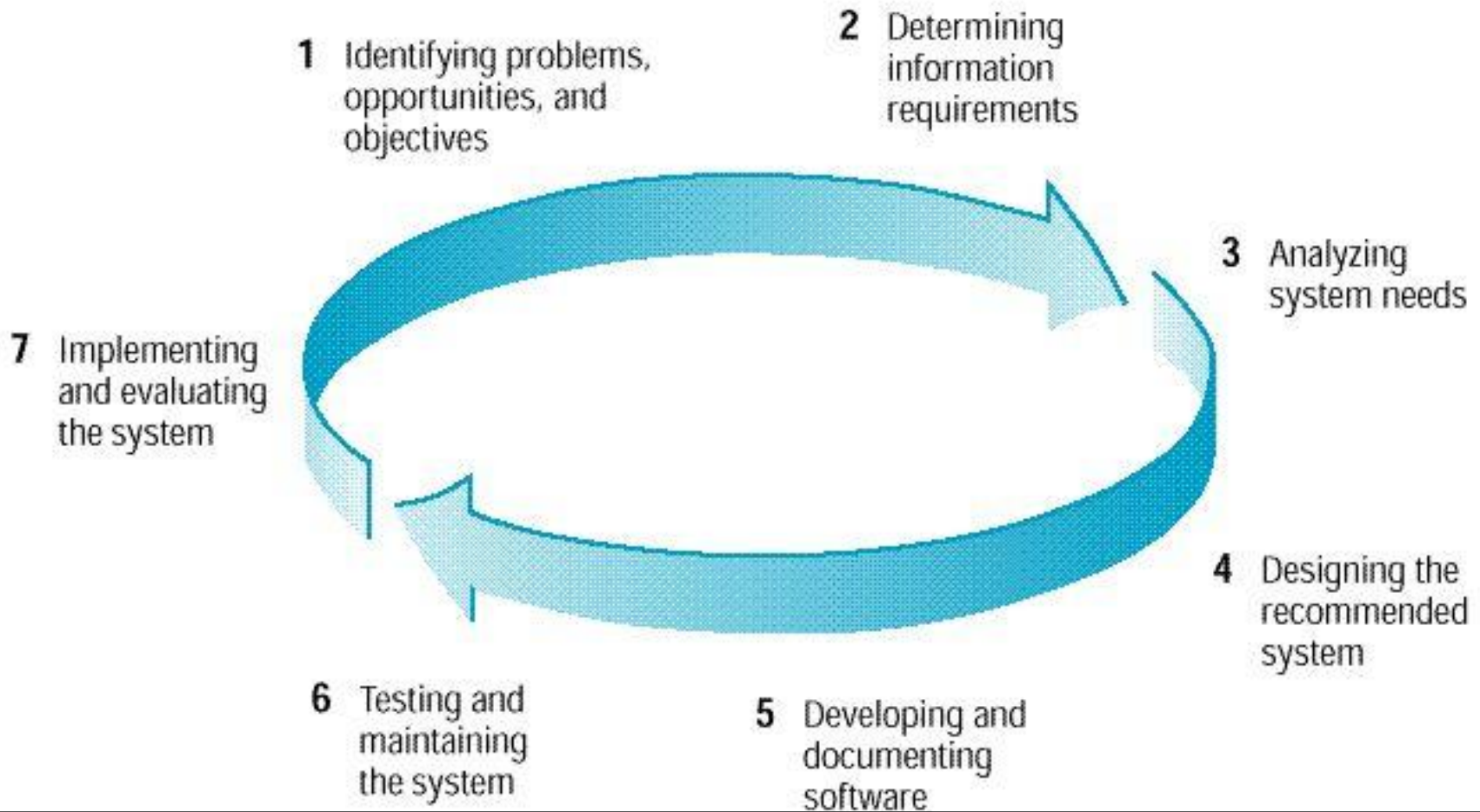
(Kendall & Kendall terminology)







**Figure 1.2 The Seven Phases of the Systems Development Life Cycle**



# Suggested Activities

## Review of pre-requisite topics (platform tech)

**Activity 1)** Title: Procedure for Installing and Testing the Apache Web Server on Raspberry Pi/Debian Linux Within a VirtualBox Environment

**Activity 2)** Title: Installing MariaDB/MySQL on Debian Linux in a VirtualBox Environment and Integrating with Apache and PHP

**Activity 3)** Building a CRUD Web Application on a Virtual Environment Integrating Linux, Apache, MySQL, and PHP (*Scratch PHP Development*)

# Suggested Activities

System Integration and Architecture (Activities)

**Activity 4)** Title: Detailed Discussion of a System Architecture Design of a Proposed Information System

**Activity 5)** Title: Prototype of the System Architecture using Visualization Software (Figma, Adobe XD, Axure RP etc)

# Reference books

- Sage A.P. and Rouse, W.B. Handbook of Systems Engineering and management, John Wiley & Sons, 1999.
- Kendall And Kendall, System Analysis and Design