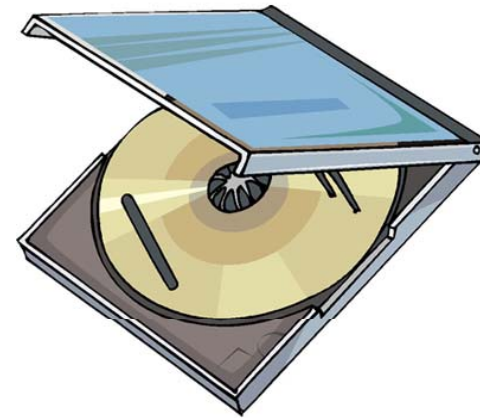# SOFTWARE SYSTEM ENGINEERING: A TUTORIAL

Richard H. Thayer

발표자 : 이동아

INTRODUCTION

- Specific of software
  - Getting Large
  - Getting complex

System

- A collection of elements related in a way that allows the accomplishment of a common object

- System Engineering(SE)

  - The practical application of scientific, engineering, and management skills necessary to transform an operational need into a description of a system configuration that best satisfies that need

- Functions of SE

  - Problem definition
  - Solution analysis
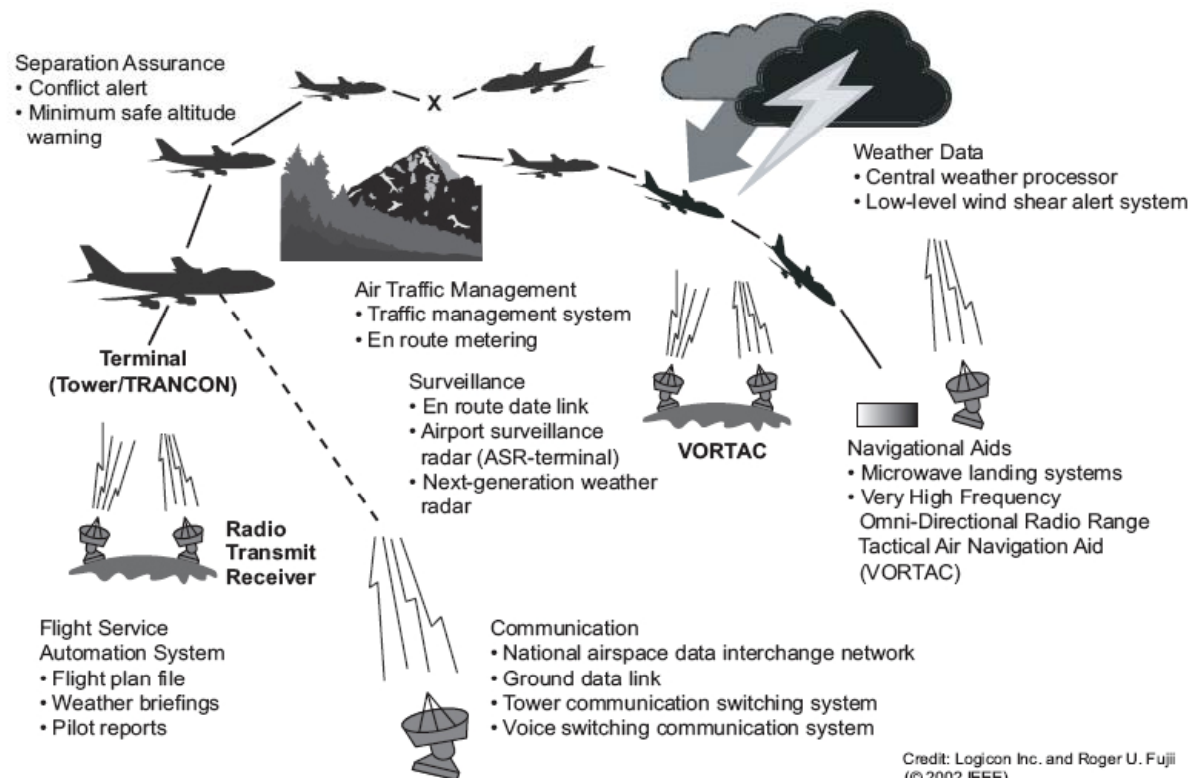 - Process planning
  - Process control
  - Product evaluation

Software System Engineering(SwSE)

- A Technical and Management process
- The technical process of SwSE is the analytical effort necessary to transform an operational need into :
  - A description of a software system
  - A software design of the proper size, configuration, and quality
  - Documentation requirements and design specifications
  - The Procedures necessary to verify, test, and accept the finished product
  - The documentation necessary to use, operate, and maintain the system.

Software System Engineering(SwSE)

□ The role of software in system

■ The cohesiveness and control of date that enables the system to solve problems

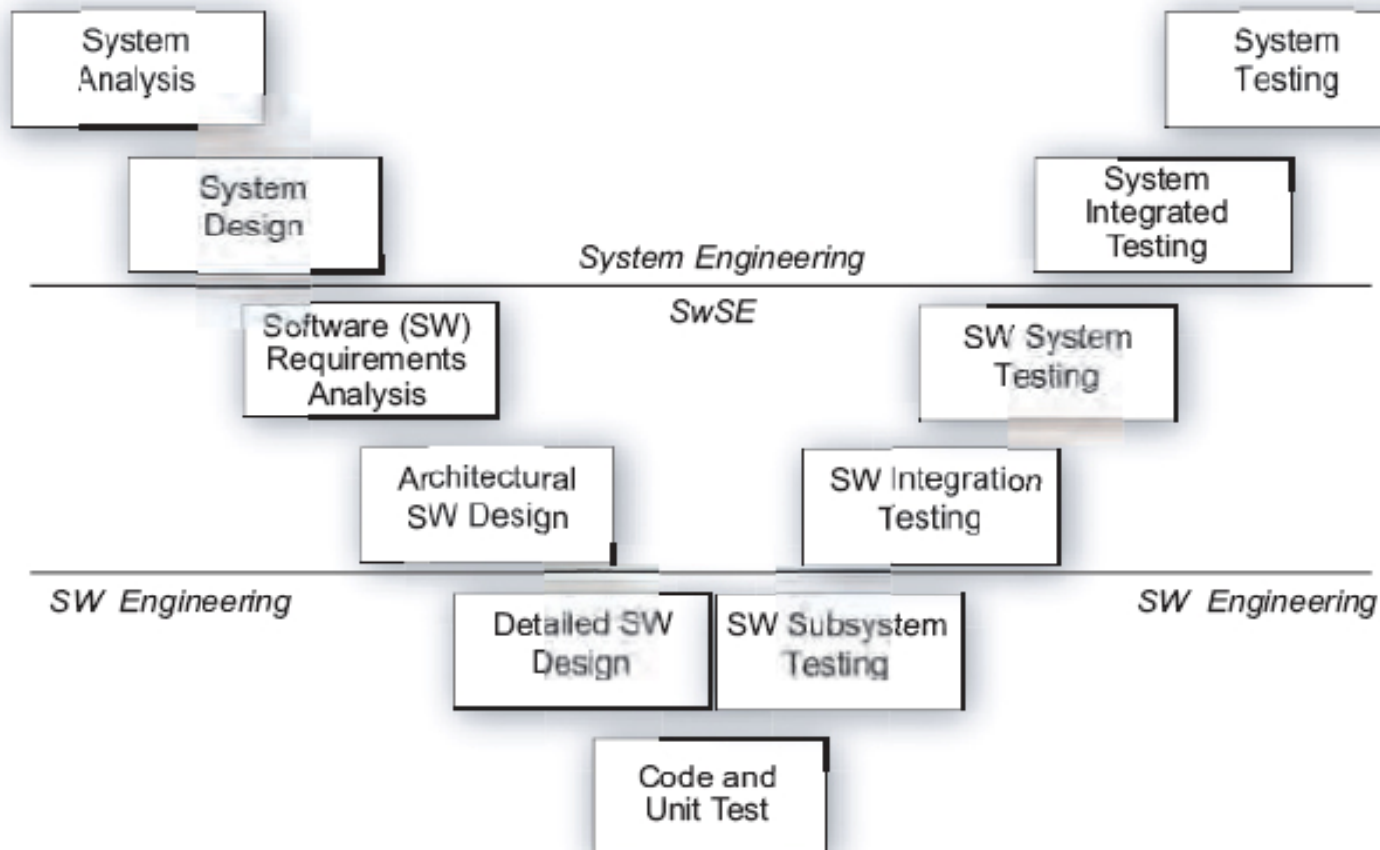■ The flexibility to work around hardware or other problems

Separation Assurance
• Conflict alert
• Minimum safe altitude warning

Weather Data
• Central weather processor
• Low-level wind shear alert system

Air Traffic Management
• Traffic management system
• En route metering

Terminal
(Tower/TRANCON)

Surveillance
• En route date link
• Airport surveillance radar (ASR-terminal)
• Next-generation weather radar

VORTAC

Navigational Aids
• Microwave landing systems
• Very High Frequency Omni-Directional Radio Range Tactical Air Navigation Aid (VORTAC)

Radio Transmit Receiver

Flight Service Automation System
• Flight plan file
• Weather briefings
• Pilot reports

Communication
• National airspace data interchange network
• Ground data link
• Tower communication switching system
• Voice switching communication system

Credit: Logicon Inc. and Roger U. Fujii
(© 2002 IEEE)

Software System Engineering(SwSE)

☐ **What Is Software Engineering(SwE)?**

- ☐ The practical application of computer sciences to the analysis, design, construction and maintenance of software and its associated documentation

- ☐ An engineering science that applies the concepts of analysis, design, coding, testing, documentation, and management to the successful completion of large, custom-built computer programs under time and budget constraints

- ☐ The systematic application of methods, tools and techniques to achieve a stated requirement or objective for an effective and efficient software system
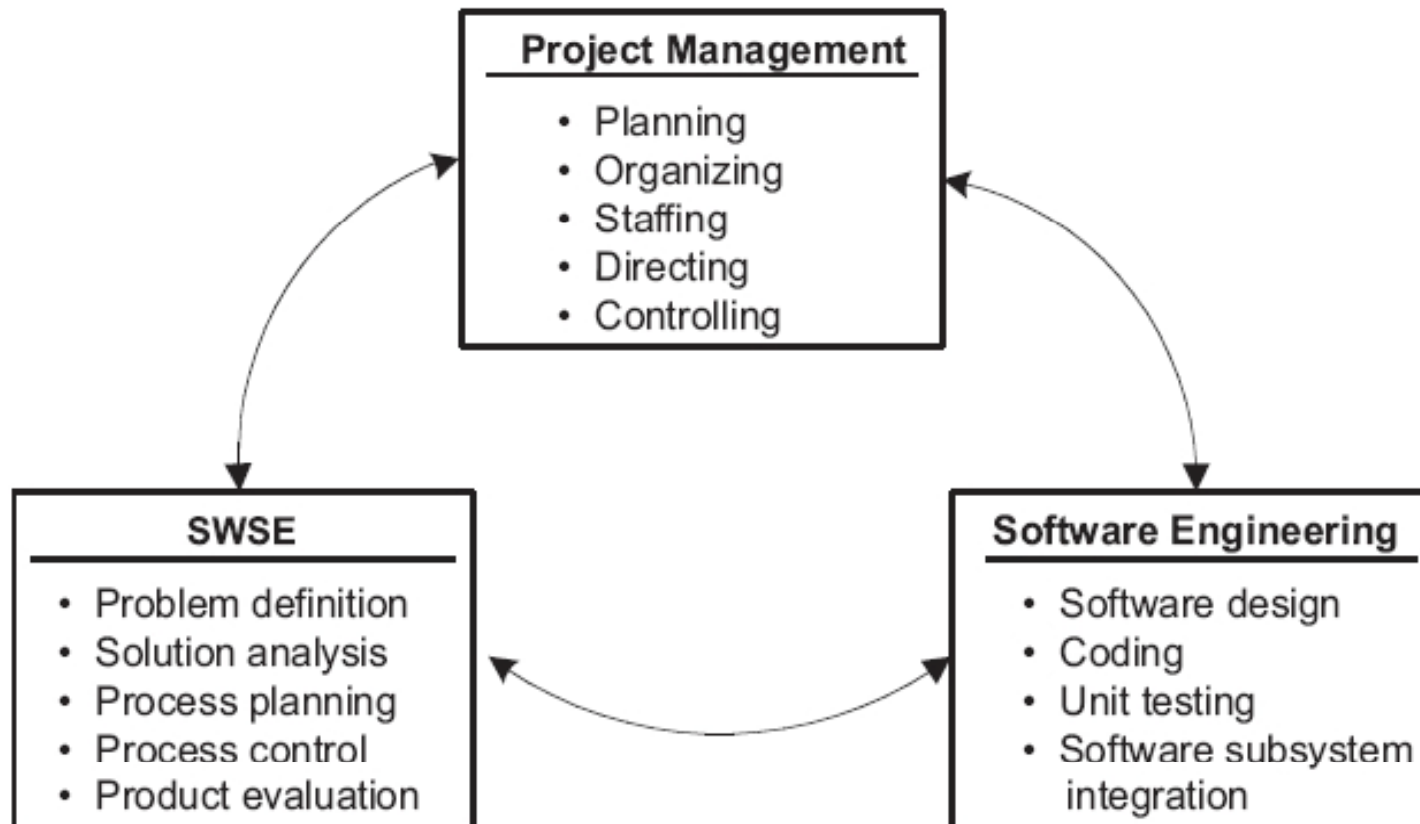
Software System Engineering(SwSE)

☐ Engineering Relationship

System Analysis		System Testing
System Design	System Engineering	System Integrated Testing
	SwSE	
Software (SW) Requirements Analysis		SW System Testing
Architectural SW Design		SW Integration Testing
SW Engineering		SW Engineering
Detailed SW Design	SW Subsystem Testing	
	Code and Unit Test	

(© 2002 IEEE)

Software System Engineering(SwSE)

☐ What Is Project Management(PM)?

Project Management
- Planning
- Organizing
- Staffing
- Directing
- Controlling

SWSE
- Problem definition
- Solution analysis
- Process planning
- Process control
- Product evaluation

Software Engineering
- Software design
- Coding
- Unit testing
- Software subsystem integration

(© 2002, IEEE)

Software System Engineering(SwSE)

□ Functions of Software System Engineering

- ▫ Requirements analysis
- ▫ Software design
- ▫ Process planning
- ▫ Process control
- ▫ Verification, validation, and testing(VV&T)

□ The overall technical management of a system development project

Software Requirements Analysis

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

- **Functional requirement**

- **Performance requirement**

- **External interface requirement**

- **Design constraint**

- **Quality attribute**

Software Requirements Analysis

- **Functional requirement**
  - specify functions that a system or system component must be capable of performing

- **Performance requirement**
  - specify performance characteristics that a system or system component must possess such as speed, accuracy, and frequency

- **External interface requirement**
  - Specify hardware, software, or database elements with which a system or component must interface, or set forth constraints on formats, timing, or other factors caused by such an interface

- **Design constraint**
  - Affect or constrain the design of a software system of software system component

- **Quality attribute**
  - Specify the degree to which software possesses attributes that affect quality, such as correctness, reliability, maintainability, and portability

Software Requirements Analysis

Initiation	▪Acquirer and user system requirements have been properly identified ▪A top-level system design is complete ▪A set of software subsystem is identified ▪All user requirements have been properly allocated(assigned) to one or more of these subsystems
Completion	▪A concept of operations(ConOps) document is complete ▪All of the software system requirements are identified ▪An SRS* that identifies "What" the system must do is written ▪The SRS is verified ▪A requirements tracing is complete ▪A preliminary user manual is written ▪A test compliance matrix is finished ▪A preliminary test specification is finished ▪The software specification review(SSR) is complete ▪A requirements baseline is established

*Software requirements specification

Software Design(Solution Analysis)

- The process of selecting and documenting the most effective and efficient system elements

- Architectural design
  - Equivalent to system design

- Detailed design
  - Equivalent to what SE calls "component engineering," and is considered part of the software engineering phase

Software Design(Solution Analysis)

Initiation	▪The software requirements have been properly identified and documented ▪The draft user's manual and draft test documents have been developed
Completion	▪The "how" question is answered(conceptual design) ▪The architectural description is complete ▪An architectural design review is complete and the stakeholders have accepted the design description ▪The preliminary operator's and maintenance manuals are written ▪The design tracing is complete ▪The architectural design is reviewed ▪The architectural design is verified ▪The product baseline is established

Process Planning

□ Specify the project goals and objectives and the strategies, policies, plans, and procedures for achieving them

□ Process Planning vs Project Planning

Software system engineering determines :	Project management determines :
The tasks to be done	The skills necessary to do the task
The order of precedence between tasks	The schedule for completing the project
The size of the effort(in staff time)	The cost of the effort
The technical approach to solving the problem	The managerial approach to monitoring the project status
The analysis and design tools to use	The planning tools to use
The technical risks	The management risks
The process model to be used	The process model to be used
Update to the plans when requirements or development environments change	Updates to the plans when managerial conditions and environments change

Process Control

□ A feedback system that provides information on how well the project is going

□ Process Control vs Project Control

Software system engineering:	Project management:
Determines the requirements to be met	Determines the project plan to be followed
Selects technical standards to be followed, for example, IEEE Std. 830	Selects managerial standards to be followed, for example, IEEE Std. 1058
Establishes technical metrics to control progress, for example, requirements growth, errors reported, or rework.	Establishes management metrics to control progress, for example, cost growth schedule slippage, or staffing shortages
Uses peer reviews, in-process reviews, software quality assurance, VV&T, and audits to determine adherence to requirements and design	Uses joint acquirer-developer (milestone) reviews and software configuration management to determine adherence to cost, schedule, and progress
Reengineers the software requirements when necessary	Restructures the project plan when necessary

VV&T(Product Evaluation)

- Verification
  - determines whether the products of a given phase of the software development cycle fulfill the requirements established during the previous phase. Verification answers the question, *"am I building the product right?"*

- Validation
  - determines the correctness of the final program or software with respect to the user's needs and requirements. Validation answers the question, *"am I building the right product?"*

- Testing
  - the execution of a program or partial program, with known inputs and outputs that are both predicted and observed for the purpose of finding errors. Testing is frequently considered part of validation

Summary and Conclusions

- Conducting SwE without conducting SwSE puts a project in jeopardy

- SwE and SwSE are primarily disciplines used in the front end of the system life cycle and at the latter part of the life cycle

- The other major activity of SwSE is the final validation and testing of the completed system

- SwSE is not cheap, but it is cost effective