Software Engineering
CS5704: Class 6 - 2/23/01

Instructor: Shawn A. Bohner
Voice: (703) 538-8374
Email: bohner@nvc.cs.vt.edu

Teaching Assistant: Sepna Georges
Voice: (703) 538-8381
Email: sgeorge@vt.edu

Agenda

▲ Review Last Week’s Material
● Turn in Homework
● Discussion
▲ Chapter 10 – System Engineering
▲ Chapter 11 – Analysis Concepts and Principles
● Break
▲ Mid-Term Review
▲ Homework Assignments
Problem 9.1


▲ “No matter where you are in the system life cycle, the system will change, and the desire to change it will persist throughout the life cycle.”

▲ Since change is a fact of life in software, it is necessary to recognize that iteration occurs in all paradigms for software engineering.

▲ All software paradigms must accommodate change. The biggest danger is to assume that the software engineering process will flow sequentially from start to finish with no changes. This just is not realistic!
Problem 9.2

Objective: Discuss Reasons for Baselines

Baselines establish an inflection point where an aggregate of changes can be approved. We must establish a point at which we "cut the chord."

We must define a point beyond which we will not change something without careful (even formal) evaluation and approval – create another baseline.

Problem 9.3

A small project might combine analysis and design modeling into a "Development Specification" that would serve as the first SCI. The source would be the second. It is unlikely that a formal Test Specification would be created, but the suite of test cases would be the third SCI. A User's manual would likely be the fourth SCI and executable versions of the software would be the fifth.

Any change to these baseline SCIs might generate the following questions:
1. What are the effort and cost required?
2. How complex is the change and what is the technological risk?
3. Is software to be changed heavily coupled to other system components?
4. Are the modules to be changed cohesive?
5. What is likely to happen if the change is not implemented properly?
6. What is the importance of the change compared with other changes?
7. Who will make the change?
8. How can we be sure that the change is implemented properly?
Problem 9.8

▲ Objective: <part of> & <interrelated>;
describe 5 more relationships useful in project database

<depends on>              <traces to/from>
<mapped from>             <describes>
<derived from>            <uses>
<model of>                <created using>
<identified as>           <approves>

Problem 9.12

▲ Objective: Compare SCM audit to Formal Technical Review. Can they be combined? Pros and Cons.

▲ SCM audit focuses on compliance with software engineering standards criteria
▲ FTR concentrates on discovering errors associated with function, performance or constraints
▲ SCM audit has a less technological perspective while FTRs are technical communications.
▲ Audits more often conducted by SQA groups
▲ Pros – time, same project staff
▲ Cons – lose focus, separation of concerns
Key Relationships and Dependencies

Business Context

Business Drivers

Enterprise Business Architecture
- Strategies
- Processes

Corporate Resources & Organization

Technical Architecture
- Frameworks
- Components

Information Architecture

Applications Portfolio
- Custom
- Packaged
- Package/Custom

IT Infrastructure

IT Resources
Program/Portfolio Management
Staffing Profile

What are some examples of the above?

Change Management and SCM

▲ Changes starts somewhere
▲ Must be associated with the software configuration and managed through the change cycle

Demand Management

Configuration Management

What is Demand Management? Release Management?
What impact do they have on Configuration Management?
What are the Changes?

Changes in
Business Requirements
Changes in
Technical Requirements
Changes in
User Requirements

Software Models

Project Plan

Test

Data

Code

Other Documents

What is an example of a change that impacts all of the above?

Flow of SCM: Definition, Use, Archive

SCM Staff

- Defines SCM Structures, Standards and Procedures

Software Project Staff

- Uses SCM Library Structures, Standards and Procedures

SCM Staff

- Archives/Controls Baseline Software

- Key Software Baseline Versions
- Archives of System Software

Why do we baseline the software?
At what stages will software be baselined?
Chapter 10
System Engineering

Purpose

▲ The purpose of this chapter is to provide an introduction to the process of system engineering
  ● The emphasis is on requirement analysis, specification, and quality management
  ● Simple examples of system modeling are presented
▲ Objectives
  ● Outline key system engineering principles
  ● Show process first and information first approaches
  ● Examine key aspects of software in the context of a system and how to model them
**Business Engineering**

- **Vision**
- **Corporate Culture**
- **Information Technology**
- **Business Process**

**Engineer the Business**

- **People, Technology**
- **Market Demand, Capital, Competition, Regulations**
- **Current Business**
- **Engineered Business**

**Business Process Engineering (BPE)**

**Engineering Objectives**

- Understand Existing Processes
- Determine Process(es) to be (Re)Designed
- Design Process Alternatives
- Pilot Newly Selected Process(es)
- Implement New Process(es)

**Current Business Process**

**Engineer the Business Process**

**New Business Process**

**BPE Tools**
The Decomposition Hierarchy

▲ World view – most abstract
▲ Detailed view – almost atomic
▲ Systems are made of systems
  ● Like the economy
▲ Analyze systems to understand them
▲ Make it simple but not too simple – Albert Einstein

Why Model the Process or System?

▲ Understand the process or system
  ● Modeling "answers questions"
▲ Arrive at Consensual Reality
  ● Common view of what is going on and what the "vision" is (often for the first time)
▲ See opportunities to improve by recognizing patterns and analyzing:
  ● Disconnects and overloads
  ● Ambiguous responsibilities
  ● Duplications, long cycle times and other inefficiencies
▲ Establish a visible way to measure and improve
Cycle of Continuous Improvement

Define System
- Record
- Validate
- Assess

Change System
- Adjust
- Confirm
- Automate

Control System
- Instrument
- Measure
- Analyze

Stabilize System
- Document
- Disseminate
- Institutionalize

Process / System Modeling – Activities

- Representation Approach: Structured Analysis & Design Technique (SADT or IDEF0) -- technique from system engineering
  - Graphic Language
  - Graph/Text page pairs
  - Model also contains purpose, viewpoint, and context

Verb Phrase (action) → noun Phrase (thing)

Controls, Constraints

Inputs Required → Activity → Outputs Produced

Mechanisms, Responsibilities
Decomposition

- Page pairs of diagrams and descriptive text
- Goal: two new readers will have the "same" understanding and interpretation
- Decompose until purpose of model is achieved (i.e., can answer predetermined questions with adequate precision)

Context for Process Model

Purpose: To model the current Carrier Access Billing System (CABS) Software Maintenance Process
- Identify and describe all major resource consuming activities in CABS
- Identify improvement opportunities (e.g., sources and sinks of time, effort, cost, quality, and risk) that can be measured and monitored over time

Viewpoint: Senior Programmer Analyst
Context: CIS Business Environment
Process Modeling – Interactions

- Vertical bars are organizations
  - Customer, functional group, person
- Horizontal arrows are information or objects
  - Products, items
  - Approvals, status

A0 Diagram of Maintain, Enhance CABS
BPE via Information Engineering

▲ Uses integrated set of procedures, methods, and tools to identify how information systems can best meet the strategic goals of an enterprise
▲ Focuses first on the enterprise and then on the business area
▲ Creates enterprise models, data models and process models
▲ Creates a framework for better information management distribution, and control
▲ Just like Engineering Systems... Businesses are Systems too!

BPE Major Activities

▲ Information Strategy Planning (ISP)
  ● strategic goals defined
  ● success factors/business rules identified
  ● enterprise model created
▲ Business Area Analysis (BAA)
  ● processes/services modeled
  ● interrelationships of processes and data
▲ Application Engineering
  ● a.k.a ... software engineering
  ● modeling applications/procedures that address BAA and constraints of ISP
▲ Construction and Delivery
  ● using CASE and 4GTs, testing
Information Strategy Planning

▲ Management Issues
- Define strategic business goals/objectives
- Isolate critical success factors
- Conduct analysis of technology impact
- Perform analysis of strategic systems

▲ Technical Issues
- Create a top-level data model
- Cluster by business/organizational area
- Refine model and clustering

Defining Goals and Objectives

▲ Goal—Statement of Direction
- “Reduce manufactured cost of product”

▲ Objective defines Measurable Goal
- Decrease reject rate by 20% in first 6 months
- Gain 10% price concessions from suppliers
- Reengineer 30% of components for ease of manufacture during first year

▲ Goals tend to be Strategic while Objectives tend to be Tactical
Business Area Analysis (BAA)

▲ groupings of business functions and data” (Martin)
▲ Perform many of the same activities as ISP, but narrow scope to individual business area
▲ Identify existing information systems / determine compatibility with new ISP model
  ● define systems that are problematic
  ● defining systems that are incompatible with new information model
  ● begin to establish re-engineering priorities

The BAA Process

- Process Flow Models
- Data Model
- Process Decomp. Diagram
- Matrices e.g., entity/ process matrix
Product Engineering

▲ Software is an important part of the overall system... but only a part
▲ Software is not the business... it enables the business
▲ Software must be analyzed and evolved in the context of the business

Requirements Engineering

▲ Elicitation — determining what the customer requires
▲ Analysis & negotiation — understanding the relationships among various customer requirements and shaping those relationships to achieve a successful result
▲ Requirements specification — building a tangible model of requirements
Requirements Engineering (continued)

▲ System Modeling — building a representation of requirements that can be assessed
▲ Validation — reviewing / assessing the model for correctness, completeness, and consistency
▲ Management — identify, control and track requirements and the changes that will be made to them

Basic Product Architecture Template

user interface processing

input processing | process and control functions | output processing

maintenance and self-test
Chapter 11
Analysis Concepts and Principles
Purpose

▲ The purpose of this chapter is to describe the process of software requirements analysis as a refinement of the initial work products developed during the systems engineering process
  ● Complete a system specification document before writing software specification
  ● Focus of software requirements analysis is on working with the customer to identify and refine the information, functional, and behavioral requirements
▲ Objectives
  ● Outline key software requirements analysis principles
  ● Examine key aspects of software modeling

Software Requirements Analysis

▲ Identify the “customer” and work together to negotiate “product-level” requirements
▲ Build an analysis model
  ❑ Define function
  ❑ Represent behavior
  ❑ Focus on data
▲ Prototype areas of uncertainty
▲ Develop a specification that will guide design
▲ Conduct formal technical reviews
Requirements Gathering

Facilitated Application Specification Techniques (FAST)

Software Engineering Group

Customer Group

Anyone seen any chairs around here?

FAST Guidelines

▲ Participants must attend entire meeting
▲ All participants are equal
▲ Preparation is as important as meeting
▲ All pre-meeting documents are to be viewed as “proposed”
▲ Off-site meeting location is preferred
▲ Set an agenda and maintain it
▲ Don’t get mired in technical detail

J. Wood & D. Silver
Quality Function Deployment

▲ Function deployment determines the “value” (as perceived by the customer) of each function required of the system
▲ Information deployment identifies data objects and events
▲ Task deployment examines the behavior of the system
▲ Value analysis determines the relative priority of requirements

Use-Cases

▲ A collection of scenarios that describe the thread of usage of a system
▲ Each scenario is described from the point-of-view of an “actor”—a person or device that interacts with the software in some way
▲ Each scenario answers the following questions:
  ● What are the main tasks of functions performed by the actor?
  ● What system information will the actor acquire, produce or change?
  ● Will the actor inform the system about environmental changes?
  ● What information does the actor require of the system?
  ● Does the actor wish to be informed about unexpected changes
The Analysis Process

Five Analysis Principles

Analysis Principle 1  Model the Data Domain
- Define data objects
- Describe data attributes
- Establish data relationships

Analysis Principle 2  Model the Functions
- Identify functions that transform data objects
- Indicate how data flow through the system
- Represent producers and consumers of data

Analysis Principle 3  Model the Behavior
- Indicate different states of the system
- Specify events that cause the system to change state

Analysis Principle 4  Partition the Models
- Refine models to represent lower levels of abstraction
- Refine data objects
- Create a functional hierarchy
- Represent behavior at different levels of detail

Analysis Principle 5  Essence
Begin by focusing on the essence of the problem without regard to implementation details
Al Davis’ Operational Analysis Principles

▲ Understand the problem before you begin to create the analysis model.
▲ Develop prototypes that enable a user to understand how human-machine interaction will occur.
▲ Record the origin of and the reason for every requirement.
▲ Use multiple views of requirements.
▲ Prioritize requirements.
▲ Work to eliminate ambiguity.

The Analysis Model

Data Model

Functional Model

Behavioral Model
Homework Assignment for 3/2/01

▲ Study hard for a challenging Midterm Exam
  ● Review Chapters 1-11 of SEPA text and SEPA website
  ● Class slides and notes with homework answers
  ● Assigned papers
▲ Have a great week!

▲ Some of the Homework Assignment for 3/16/01
  ● Read Chapters 13 and 14