Systems Analysis and Design With UML 2.0
An Object-Oriented Approach, Second Edition

Chapter 1: Introduction to Systems Analysis and Design

Alan Dennis, Barbara Wixom, and David Tegarden
© 2005
John Wiley & Sons, Inc.
All rights reserved. Reproduction or translation of this work beyond that permitted in Section 117 of the 1976 United States Copyright Act without the express written permission of the copyright owner is unlawful.

Request for further information should be addressed to the Permissions Department, John Wiley & Sons, Inc.

The purchaser may make back-up copies for his/her own use only and not for redistribution or resale.

The Publisher assumes no responsibility for errors, omissions, or damages, caused by the use of these programs or from the use of the information contained herein.
INTRODUCTION

Chapter 1
Key Ideas

- Many failed systems were abandoned because analysts tried to build wonderful systems without understanding the organization.
- The primarily goal is to create value for the organization.
Key Ideas

- The **systems analyst** is a key person analyzing the business, identifying opportunities for improvement, and designing information systems to implement these ideas.
- It is important to understand and develop through practice the skills needed to successfully design and implement new information systems.
THE SYSTEMS DEVELOPMENT LIFE CYCLE
Major Attributes of the Lifecycle

The project
- Moves systematically through phases where each phase has a standard set of outputs
- Produces project deliverables
- Uses deliverables in implementation
- Results in actual information system
- Uses gradual refinement
Project Phases

- **Planning**
  - Why build the system?

- **Analysis**
  - Who, what, when, where will the system be?

- **Design**
  - How will the system work?

- **Implementation**
  - System delivery
A “Simple” Process for Making Lunch

1. Assemble Sandwich Ingredients
   - Jelly
   - Peanut Butter
   - Bread
   - Cookies

2. Build Sandwich
   - Ingredients
   - Sandwich

3. Package Lunch
   - Milk
   - Lunch
Identifying business value
Analyze feasibility
Develop work plan
Staff the project
Control and direct project
Analysis

- Analysis
- Information gathering
- Process modeling
- Data modeling
Design

- Physical design
- Architectural design
- Interface design
- Database and file design
- Program design
Implementation

- Construction
- Installation
Processes and Deliverables

<table>
<thead>
<tr>
<th>Process</th>
<th>Product</th>
</tr>
</thead>
<tbody>
<tr>
<td>Planning</td>
<td>Project Plan</td>
</tr>
<tr>
<td>Analysis</td>
<td>System Proposal</td>
</tr>
<tr>
<td>Design</td>
<td>System Specification</td>
</tr>
<tr>
<td>Implementation</td>
<td>New System and Maintenance Plan</td>
</tr>
</tbody>
</table>
SYSTEM DEVELOPMENT
Methodologies
What Is a Methodology?

- A formalized approach or series of steps
- Writing code without a well-thought-out system request may work for small programs, but rarely works for large ones.
Structured Design

- Projects move methodically from one to the next step
- Generally, a step is finished before the next one begins
Waterfall Development Method

- Planning
- Analysis
- Design
- Implementation
- System
# Pros and Cons of the Waterfall Method

<table>
<thead>
<tr>
<th>Pros</th>
<th>Cons</th>
</tr>
</thead>
<tbody>
<tr>
<td>Identifies systems requirements long before programming begins</td>
<td>Design must be specified on paper before programming begins</td>
</tr>
<tr>
<td></td>
<td>Long time between system proposal and delivery of new system</td>
</tr>
</tbody>
</table>
Parallel Development

FIGURE 1-3  A Parallel Development-based Methodology
Alternatives to the SDLC

- Rapid Application Development (RAD)
- Phased Development
- Prototyping
- Throw-Away Prototyping
Rapid Application Development

- Critical elements
  - CASE tools
  - JAD sessions
  - Fourth generation/visualization programming languages
  - Code generators
Rapid Application Development Categories

- Phased development
  - A series of versions
- Prototyping
  - System prototyping
- Throw-away prototyping
  - Design prototyping
- Agile Development
- Extreme Development
How Prototyping Works

Planning → Analysis → Design → Implementation → System prototype → Implementation → System
Throwaway Prototyping
Selecting the Appropriate Methodology

- Clarity of User Requirements
- Familiarity with Technology
- System Complexity
- System Reliability
- Short Time Schedules
- Schedule Visibility
## Criteria for Selecting a Methodology

**FIGURE 1-8  Criteria for Selecting a Methodology**

<table>
<thead>
<tr>
<th>Ability to Develop Systems</th>
<th>Structured Methodologies</th>
<th>RAD Methodologies</th>
<th>Agile Methodologies</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Waterfall</td>
<td>Parallel</td>
<td>Phased</td>
</tr>
<tr>
<td>with Unclear User Requirements</td>
<td>Poor</td>
<td>Poor</td>
<td>Good</td>
</tr>
<tr>
<td>with Unfamiliar Technology</td>
<td>Poor</td>
<td>Poor</td>
<td>Good</td>
</tr>
<tr>
<td>that are Complex</td>
<td>Good</td>
<td>Good</td>
<td>Good</td>
</tr>
<tr>
<td>that are Reliable</td>
<td>Good</td>
<td>Good</td>
<td>Good</td>
</tr>
<tr>
<td>with a Short Time Schedule</td>
<td>Poor</td>
<td>Good</td>
<td>Excellent</td>
</tr>
<tr>
<td>with Schedule Visibility</td>
<td>Poor</td>
<td>Poor</td>
<td>Excellent</td>
</tr>
</tbody>
</table>
Project Team Roles and Skills
Information Systems Roles

- Business analyst
- System analyst
- Infrastructure analyst
- Change management analyst
- Project manager
# Project Team Roles

<table>
<thead>
<tr>
<th>Role</th>
<th>Responsibilities</th>
</tr>
</thead>
<tbody>
<tr>
<td>Business analyst</td>
<td>Analyzing the key business aspects of the system</td>
</tr>
<tr>
<td></td>
<td>Identifying how the system will provide business value</td>
</tr>
<tr>
<td></td>
<td>Designing the new business processes and policies</td>
</tr>
<tr>
<td>Systems analyst</td>
<td>Identifying how technology can improve business processes</td>
</tr>
<tr>
<td></td>
<td>Designing the new business processes</td>
</tr>
<tr>
<td></td>
<td>Designing the information system</td>
</tr>
<tr>
<td></td>
<td>Ensuring that the system conforms to information systems standards</td>
</tr>
<tr>
<td>Infrastructure analyst</td>
<td>Ensuring the system conforms to infrastructure standards</td>
</tr>
<tr>
<td></td>
<td>Identifying infrastructure changes needed to support the system</td>
</tr>
<tr>
<td>Change management analyst</td>
<td>Developing and executing a change management plan</td>
</tr>
<tr>
<td></td>
<td>Developing and executing a user training plan</td>
</tr>
<tr>
<td>Project manager</td>
<td>Managing the team of analysts, programmers, technical writers, and other specialists</td>
</tr>
<tr>
<td></td>
<td>Developing and monitoring the project plan</td>
</tr>
<tr>
<td></td>
<td>Assigning resources</td>
</tr>
<tr>
<td></td>
<td>Serving as the primary point of contact for the project</td>
</tr>
</tbody>
</table>
Summary -- Part 1


- The major development methodologies:
  - Structured design
    - the waterfall method
    - Parallel development
  - RAD development
    - Prototyping (regular and throwaway)
  - Agile development
    - XP streamline SDLC
Summary -- Part 2

- There are five *major team roles*: business analyst, systems analyst, infrastructure analyst, change management analyst, and project manager.
EOC Question Chapter 1

1. Compare and contrast phases, steps, techniques and deliverables.
2. Describe the major phases in the systems development life cycle (SDLC).
3. Describe the principal steps in the planning phase. What are the major deliverables?
4. Describe the principal steps in the analysis phase. What are the major deliverables?
5. Describe the principal steps in the design phase. What are the major deliverables?
6. Describe the principal steps in the implementation phase. What are the major deliverables?
7. What are the roles of a project sponsor and the approval committee?
8. What does gradual refinement mean in context of SDLC?
9. Compare and contrast process-centered methodologies with data-centered methodologies.
10. Compare and contrast structured-design based methodologies in general to RAD-based methodologies in general.
11. Compare and contrast extreme programming and throwaway prototyping.
12. Describe the major elements and issues with waterfall development.
13. Describe the major elements and issues with parallel development.
14. Describe the major elements and issues with phased development.
15. Describe the major elements and issues with prototyping.
16. Describe the major elements and issues with throwaway-prototyping.
17. What are the key factors in selecting a methodology?
18. What are the major roles on a project team?
19. Compare and contrast the role of a systems analyst, business analyst, and infrastructure analyst.
20. Which phase in the SDLC is most important and why?