



Requirements Modeling Sequence

Call New Instruction to schedule on-site training for your team! 973.746.7010

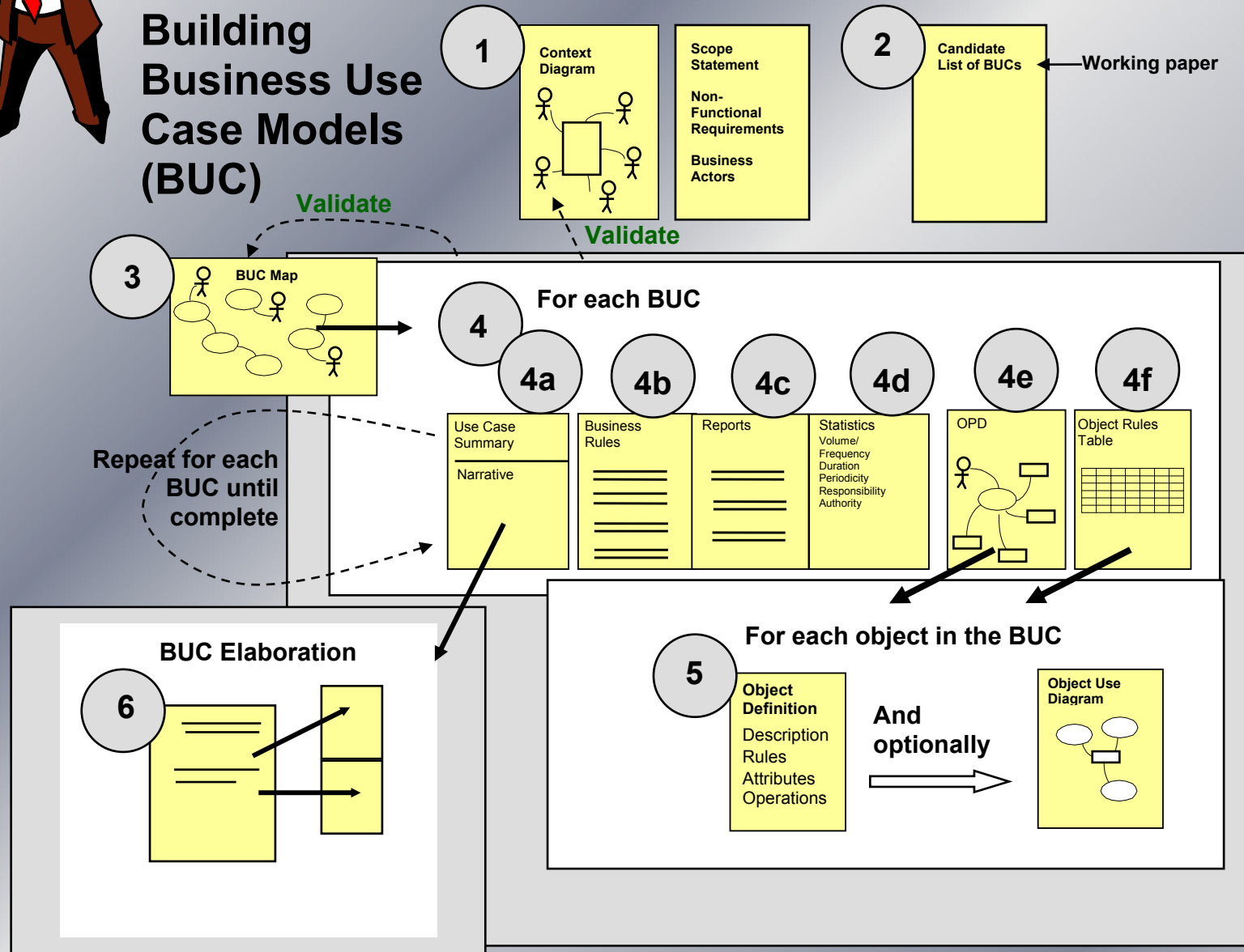


SYSTEMS & SOFTWARE RELATED WORKSHOPS

DETAILED OUTLINES AVAILABLE

- Essential Skills for the Business Analyst
- Ensuring Quality in an Agile Environment
- Implementing and Using a Methodology
- Developing Business Requirements
- A Universal Approach to Systems Analysis
- Mastering Use Case Modeling
- A Universal Approach to Systems Design
- A Universal Approach to Systems Analysis & Design
- Root Cause Analysis
- Unit Testing & System Verification
- Testing, Quality Assurance, & Security Techniques
- Analysis and Design for Managers & End Users
- Data Modeling & Warehousing Concepts
- Critical Thinker's ToolKit
- Process Mapping Bootcamp
- Object-Oriented Analysis & Design
- Mastering UML and Patterns
- UML with Rational Rose
- Project Estimating & Scheduling
- Mastering the Requirements Elicitation

Building Business Use Case Models (BUC)



1. The first step is to create a **Scope Definition** that represents the highest level view of the project.

- Identify the functions of the system necessary to meet the goals and objectives of the project.
- Identify the interactions between the system and its external world.
- Establish the expected boundaries of the system.

A **Context Diagram** shows the interaction between the project domain and the world outside the project boundaries.

The **Scope Statement** is a collection of statements that identify the aspects of the system necessary to meet the goals and objectives.

Non Functional Requirements Template includes the following:

- Performance characteristics
- Usability
- Documentation
- Error handling
- Quality
- System interfacing
- Security
- Physical environment
- Backup and recovery
- System maintenance



2. The next step is to determine the functionality of the system, this means finding the **Business Use Cases**.

- Functionality is illustrated by conceptually subdividing the context into **Business Functional Processes**.
- The Business Functional Process is diagrammed in UML with a **Business Use Case**.

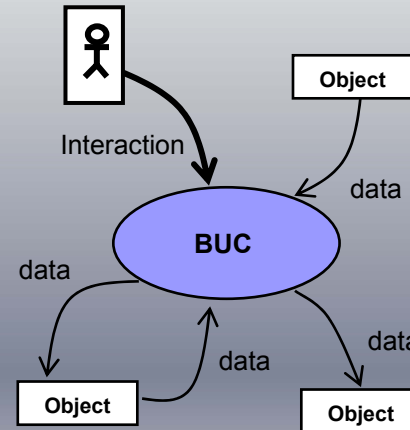
3. The next step is to create a diagram showing all of the Business Use Cases laid out in a map to indicate work flow at a very high level.

4. The next six steps get into a detailed analysis of the BUCs. Each BUC is examined in detail and modeled completely before moving on to the next one.

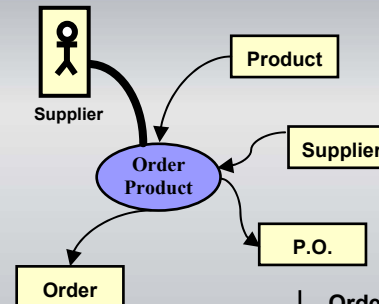
4 a. Complete the heading, summary and narrative sections of the BUC template.

4 b. c. d. Complete the BUC template with the business rules, reports and statistics.

4 e. Identify the data used and develop an **Object Participation Diagram**.



4 f. Explore the relationships among the objects as defined by the business rules and develop an object rules table.



Construct a table of the objects involved in the BUC

	Order	Product	Supplier	P.O.
Order				
Product				
Supplier				
P.O.				

5. This step focuses on each object and develops an **Object Definition** that will contain the following parts:

- Object Name
- Description
- Business Rules
- Attributes
- Operations

6. This step expands the level of detail of each BUC and factors out the common and reusable segments.

THE SOFTWARE REQUIREMENT SPECIFICATION

This is an unmistakable description of the desired behavior and performance of a system to be delivered.

The specification should include:

- Functionality – what the software is supposed to do.
- External Interfaces – interaction with people, hardware and other systems.
- Performance – speed, availability, response time, recovery time.
- Attributes – portability, correctness, maintainability, security, etc.
- Design Constraints – limitations on operating environment, resources, etc.

The specification should NOT include:

- Reference to a specific technology other than as a constraint.
- Design decisions.
- Implementation decisions.
- The availability of reusable code libraries.

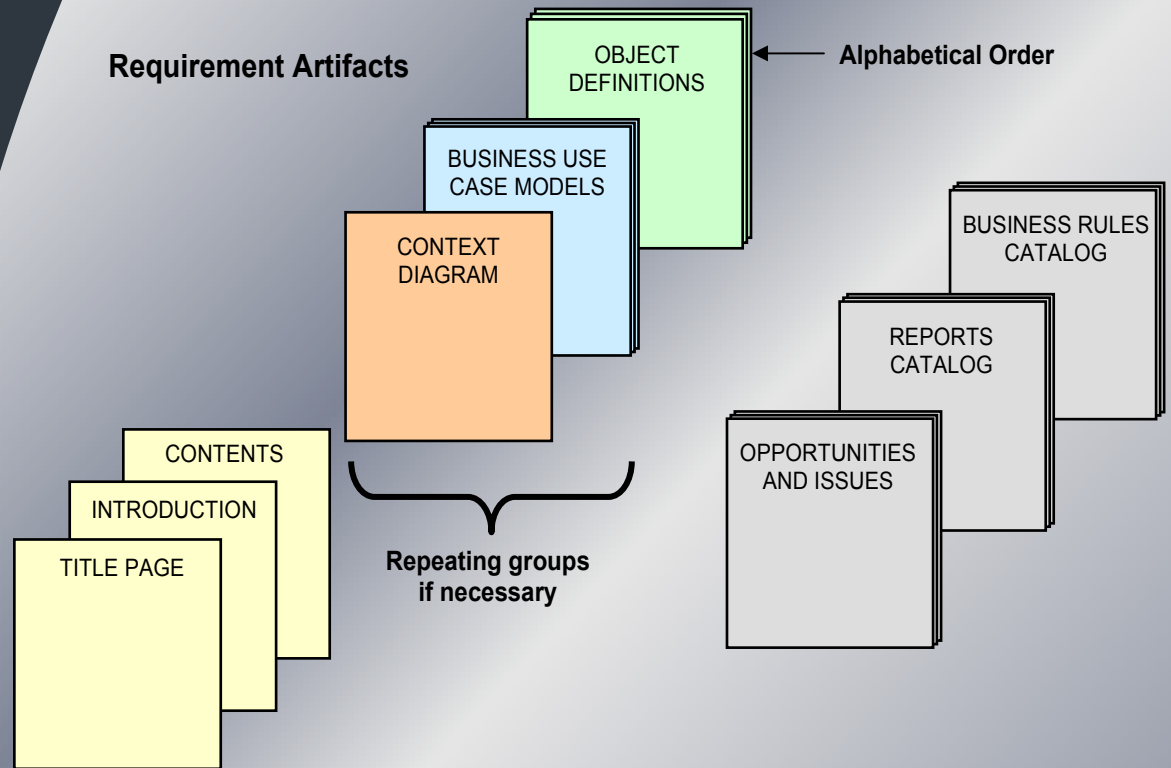
Effective requirements documents should:

- Document users needs.
- Be clearly understood by all stakeholders.
- Avoid overwhelming volume.
- Eliminate redundant statements.
- Eliminate conflicting requirements.
- Avoid premature designing.
- Ensure requirements traceability.

In good time or bad a structured requirements gathering process will save your company money.

Training from New Instruction is a SMART Investment.

Call us today to schedule training for your team!



The cost to fix errors is significantly less during the requirements phase

