“Getting the Right Information”
Improving Requirements Development & Management

Susan Kearns
ITS-ESE Systems Eng. Mgr.

Sponsored by
Michael J. Cullen,
Systems Engineering Mgr.

Introduction & Session Objectives

Objectives

- Provide managers and stakeholders an understanding of what constitutes a requirement
- Share Best Practices for gathering and managing requirements throughout the SDLC

Audience

- Project Managers
- Stakeholders
- Subject Matter Experts

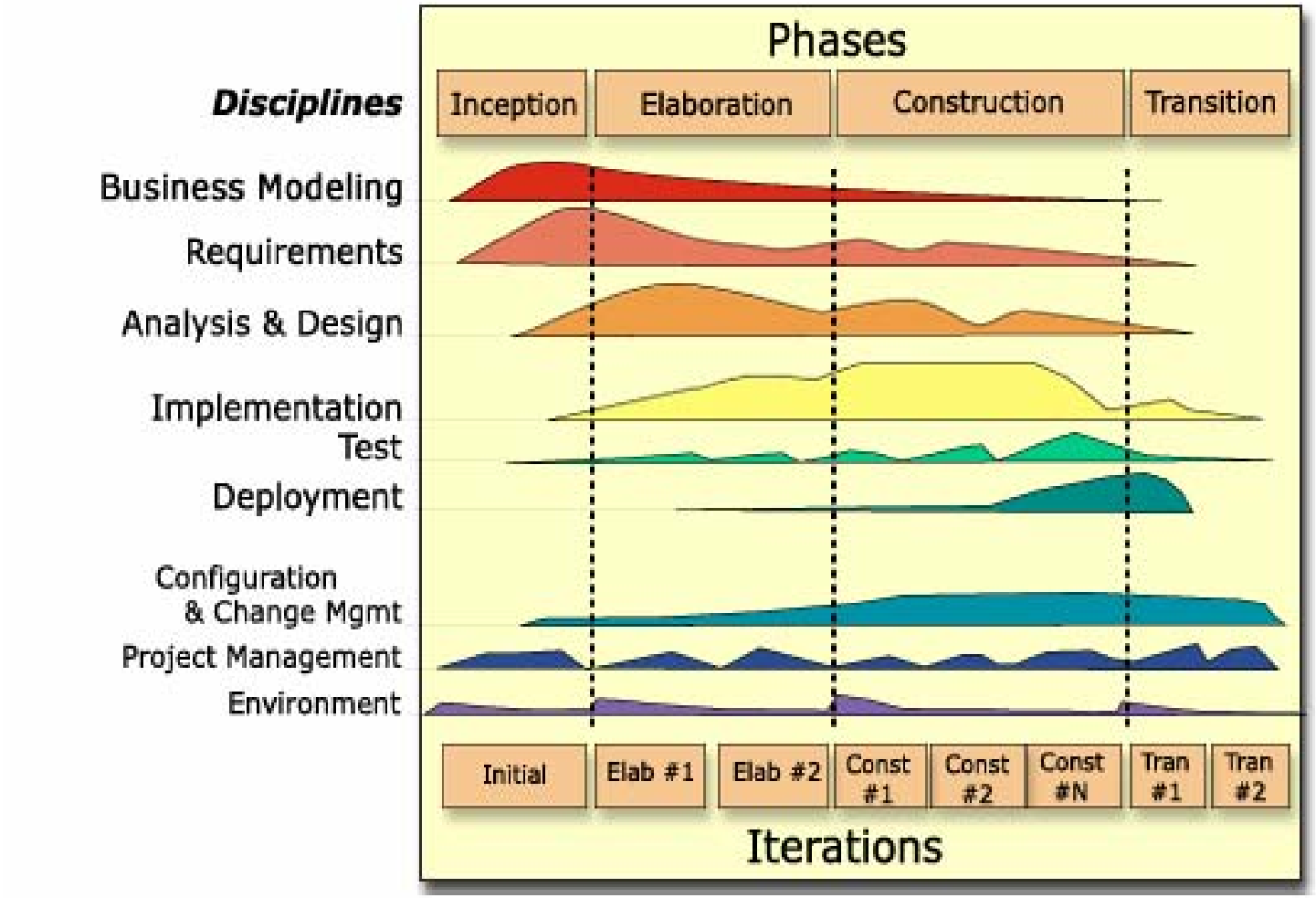
Why do IT Projects Fail?

- **IEEE Top 6**
  - **Incomplete requirements,**
  - **Lack of user involvement,**
  - **Lack of resources,**
  - **Unrealistic expectations,**
  - **Lack of executive support, and**
  - **Changing requirements and specifications**

- **"5 of 6 involve communication between builders and stakeholders"**
- **40% of IT projects fail to meet business requirements**

Source: IEEE Software, 2002; "CHAOS: The Dollar Drain of IT Project Failures", Application Development Trends, 1995 (From Standish Group research)

Where Do Requirements Fit Into The SDLC?



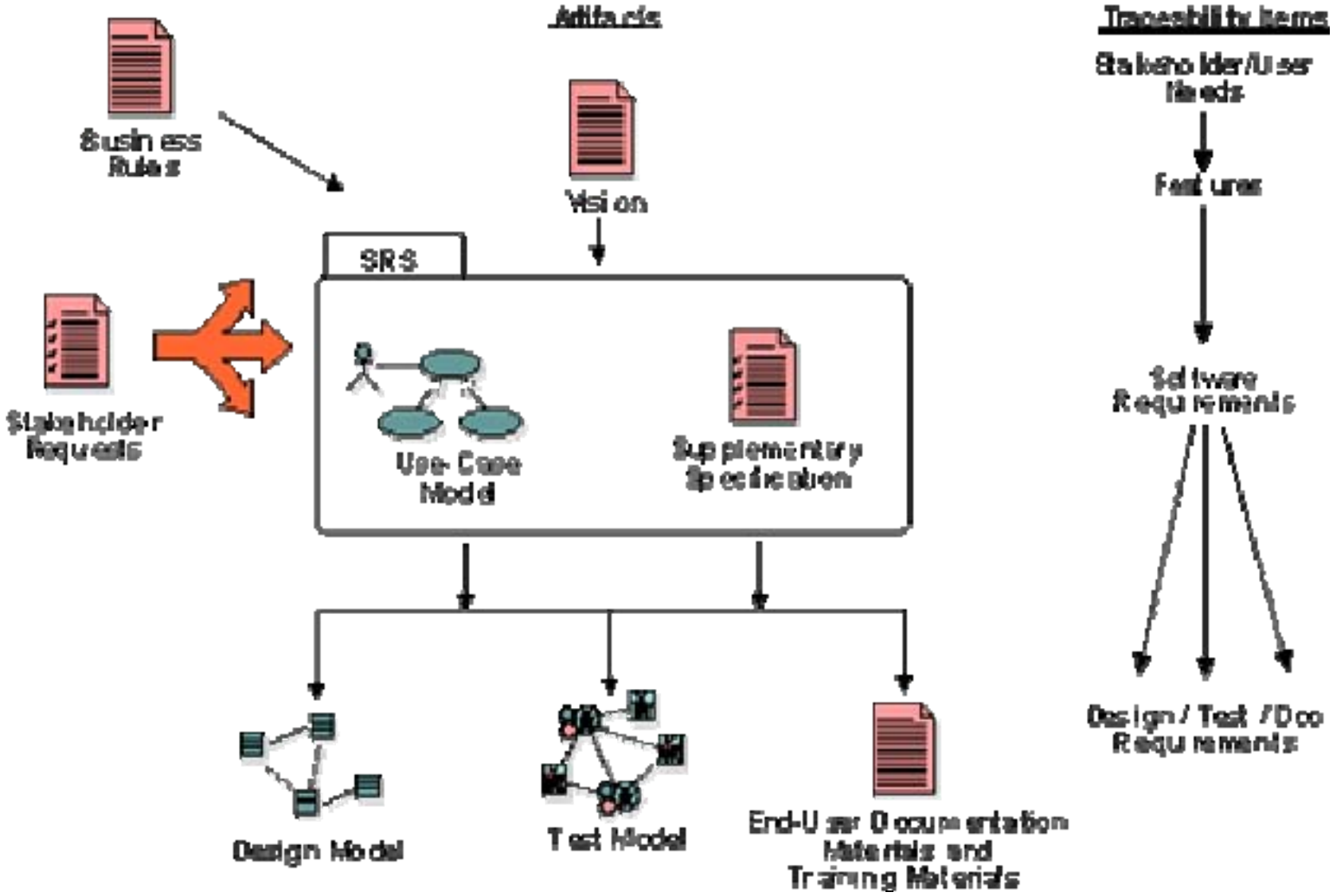
Source: IBM's Unified Process Copyright © 1987 - 2001 Rational Software Corporation

What is a Requirement?

- **A condition or capability**
  - Needed by a user to solve a problem or achieve an objective
  - That must be met or possessed by a product or product component to satisfy a contract, standard, specification, or other formally imposed document
- **A documented**
  - Representation of a condition or capability
  - Description of WHAT the system must do and WHEN it should occur
- **A basis for design**
- **Refined throughout the phases of the lifecycle**

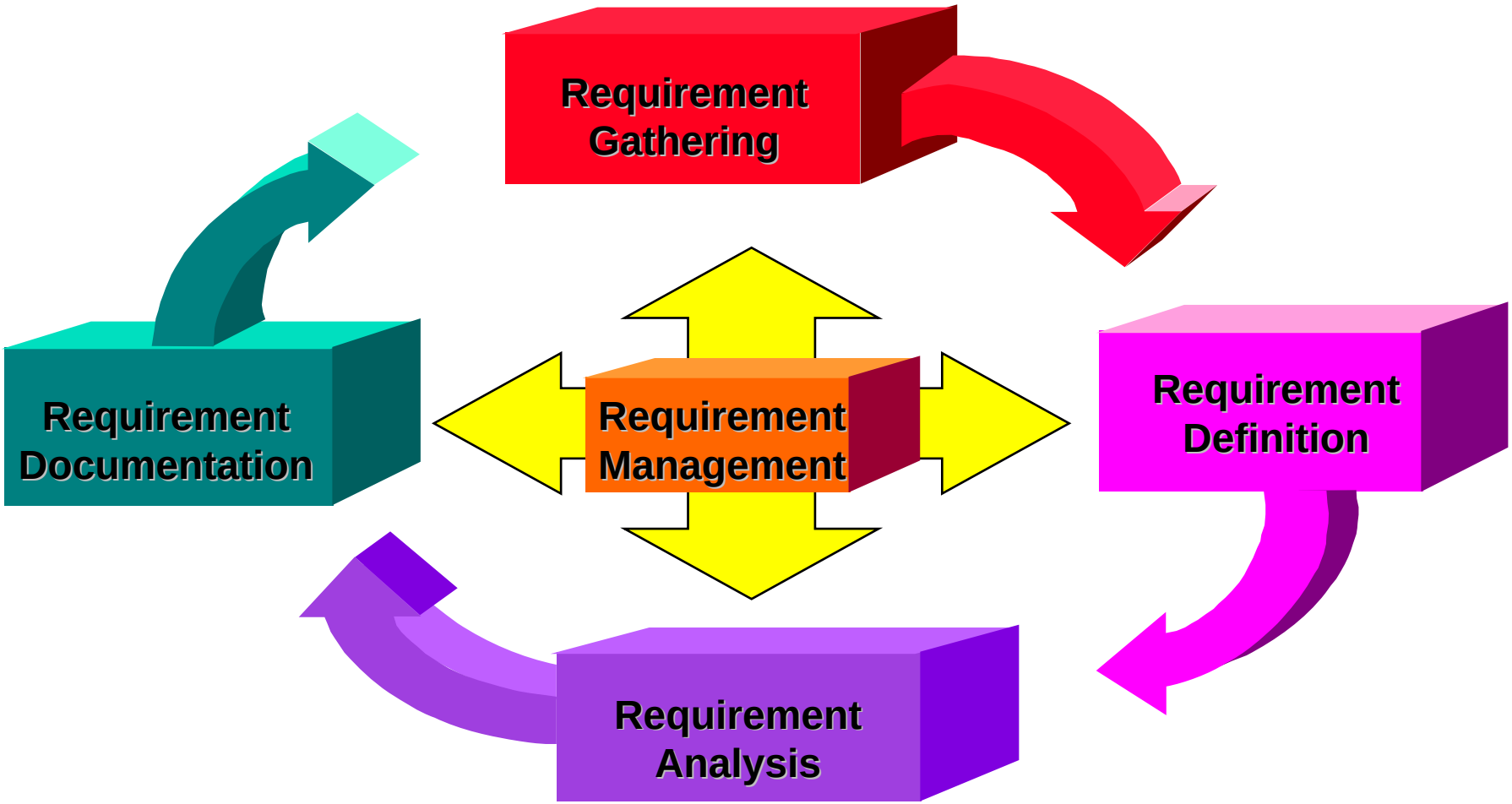
Source: Chrissis, Konrad & Shrum; CMMI – Guidelines for Process Integration & Product Improvement

The Requirements Hierarchy



Source: IBM's Unified Process Copyright © 1987 - 2001 Rational Software Corporation

Requirements Development & Management



Requirements Gathering Techniques

- Interviews
- Surveys
- Facilitated Sessions
- Use Cases, Stories & Scenarios
- Modeling
- UI Prototypes

Requirements Gathering Techniques (continued)

Interviews

- Establish rapport
- Level Set Expectations
- Capture source of information
- Differentiate between desired and critical requirements
- Best for small projects or pieces of larger projects

Surveys & Questionnaires

- Fast, easy method for broad consensus (SurveyMonkey.com)
- Samples the needs of a large population of stakeholders
- Responses are subject to interpretation
- Wording of questions can influence results

Requirements Gathering Techniques (continued)

Facilitated Sessions

- Good for consensus building
- Requires a strong facilitator
- Establish ground rules
- Decision-Makers must be present

Developing Stories, Use Cases & Scenarios

- Model system functionality or behavior
- Identify users & collaborators (Actors)
- Essential input to Analysis, Design & Test Activities
- Supplement to the System Requirements Specification

Requirements Gathering Techniques (continued)

- **Modeling**
  - Business Process (BPML)
  - Entity-Relationship (ERD)
  - Object Oriented (UML)
- **UI Prototypes**
  - Build it, drive it, change it – on the fly in the meeting
  - Captures look and feel, object behavior specification, page navigation, and type checking requirements

A Use Case Example….

Abstract

Flow

UCPU01 – Simple Search on Home Page		
Release	**Functional Area**	
1.0	Substance Search	
Description	This functionality allows users to perform a simple search for a substance from the System home page.	
Scenario	N/A	
Desired Outcome	A list of substances is generated based on the variables entered by the user.	
Actors	All Users	
Preconditions	Home Page is displayed UCPU00	
Post Conditions	Use Case AC1, AC2, AC3, UCPU03	
Authentication Required? (Y/N)	N	

Step	Action	System Response
1.	Type in either Name or CAS Number into text box and clicks the Search button.	Validate the entered data: • If no search criteria is provided, the first alternate course is applied (AC1); • If invalid data was entered, the second alternate course is applied (AC2); If data validation is successful: User is navigated to the Search Results – Substance List screen. Refer to UCPU03. • If there are no records found, the third alternate course is applied (AC3). • If there are more than 100 search results the fourth alternate course is applied. (AC4)

Requirement Number	Requirement Description
5.1.1.5	The system shall provide the capability for a user to search for a Substance by Name.
5.1.1.6	The system shall provide the capability for a user to search for a Substance by CAS Number.
5.1.1.7	The system shall display an error if the user selects “Search” without entering criteria.

Types of Requirements

- **Customer Requirements**
  - Stakeholder or User needs, expectations, constraints
  - Product features
- **Functional Requirements**
- **Non-Functional Requirements**

Usability	Reliability
Performance	Supportability
Design	Implementation
Interface	

Requirements Development Artifacts

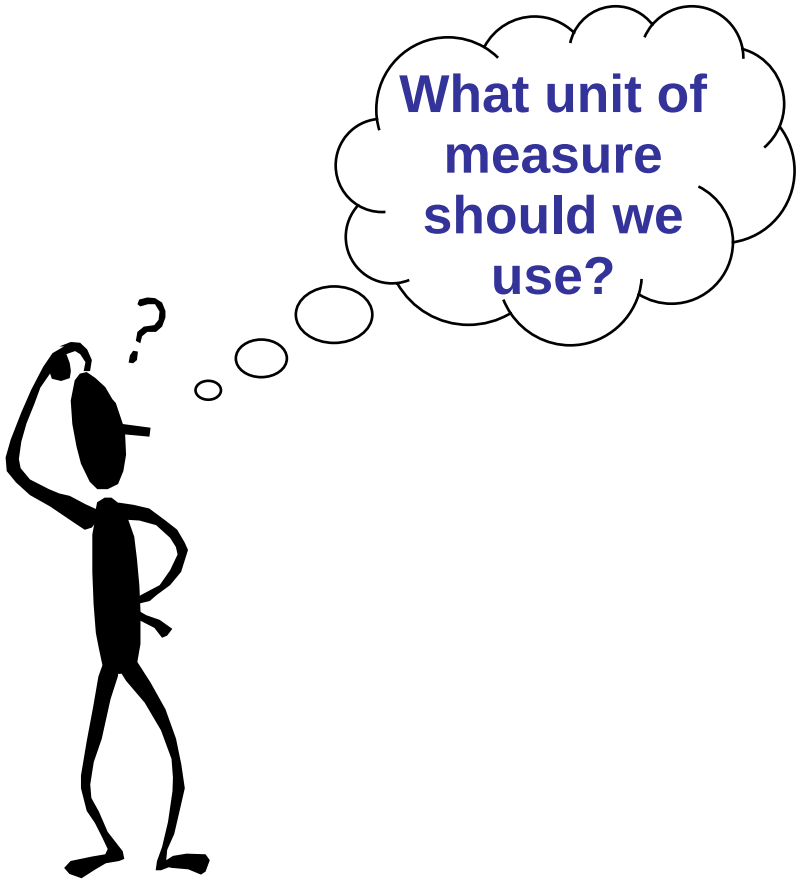
- **Inputs to Requirements Development**
  - Vision Document or Statement of Work
  - Interview Results
  - Survey Results
  - Meeting Minutes
  - Facilitated Session Minutes
  - Concept of Operations
- **Outputs of Requirements Development**
  - System Requirements Specification
  - Stories, Use Cases & Scenarios
  - Requirements Traceability Matrix

Guidelines for Writing Requirements

- Use “shall” statements
- Uniquely number requirements
- Avoid Uncertainty
- Avoid Ambiguity
- Provide the reader with sufficient notes and comments to provide context
- Consider manner of verification (analysis, test, or demonstration)

Guidelines for Requirements Avoid Uncertainty

- **Avoid TBD**
  - **To Be Determined (TBD) - a function or value that is unknown e.g., The system shall have an availability of (TBD)**
- **Avoid TBR**
  - **To Be Resolved (TBR) - a function or value that is known but may need to be refined e.g., The system shall have an availability of 0.95 (TBR)**

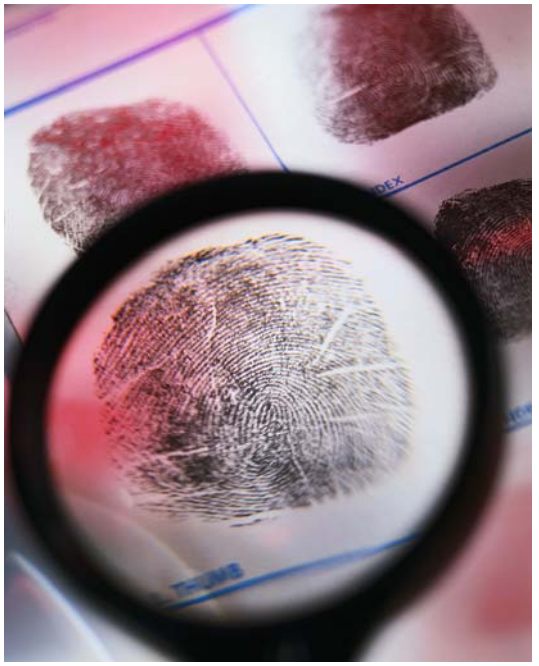


Guidelines for Requirements
Avoid Ambiguity

- Terms that are subjective and not verifiable
  - Minimize
  - Maximize
  - Rapid
  - User-friendly
  - Easy
  - Sufficiently
  - Adequate
  - Intuitive
  - Timely
  - Quick
  - Best
  - Optimize
  - Possible
  - Simultaneously
  - Sometimes
  - Suitable

Characteristics of "Good" Requirements

- ✓ Clear
- ✓ Concise
- ✓ Complete
- ✓ Consistent
- ✓ Verifiable (Testable)



- ✓ Stated in Natural Language
- ✓ Traceable

Requirements - Example

- **The system shall display a list of Substance Types that a user may search upon**

- **Is this requirement:**
  - **Clear**
  - **Concise**
  - **Complete**
  - **Consistent**
  - **Verifiable**

Source: EPA Data Standards Branch, Substance Registry Requirements Traceability Matrix

Requirements – Example (continued)

- The system shall have a Substance Search function for the following Substance Types:
  - All (default selected value)
  - Biological
  - Chemicals
  - Physical Properties
  - Miscellaneous Objects
  - Not Known

- Is this requirement:
  - Clear
  - Concise
  - Complete
  - Consistent
  - Verifiable (Testable)

Source: EPA Data Standards Branch, Substance Registry Requirements Traceability Matrix

Requirements Management

- **Managing Scope**
  - Required vs. "Nice-to-Have"
  - Prioritization
  - Traceability
- **Request Walk-Through Sessions**
  - Formalize Authorization to Proceed
- **Change Management**
  - Assess Overall Project & Schedule Impact
  - Change Request Tracking
  - Change or Configuration Control Board (CCB)
- **Risk Management**
- **Issue Tracking**

Questions

Contact Information

Session Sponsor: Michael J. Cullen
Office of Environmental Information
Cullen.Mike@epamail.epa.gov

Presenter: Susan M. Kearns
Lockheed Martin
ITS-ESE Systems Engineering Manager
Susan.M.Kearns@lmco.com