The Method of Requirements Traceability Analysis for Third-Party Auditors

May 6~8, 2026

Jang-Yeol Kim

Korea Atomic Energy Research Institute

- Table of Contents -

Requirement Traceability vs 3C+T

- Traceability analysis is the process of verifying the relationship between requirements and test results, and 3C+T refers to the **core quality criteria**– accuracy, completeness, consistency, and traceability–that requirements and verification deliverables must meet.

Checklist for 3C+T

- **Correctness**: Are the requirements correct?
- **Completeness**: Are all requirements included?
- **Consistency**: Are the requirements consistent with one another?
- **Traceability**: Are the requirements linked to the design, implementation, and testing?

Forward traceability, Backward traceability : Bi-direction

- **Two-way traceability** refers to the ability to trace the relationship between requirements and their corresponding design, code, and tests both forward and backward.
- In other words, it is a traceability method that verifies both the forward and reverse directions, rather than looking in only one direction.

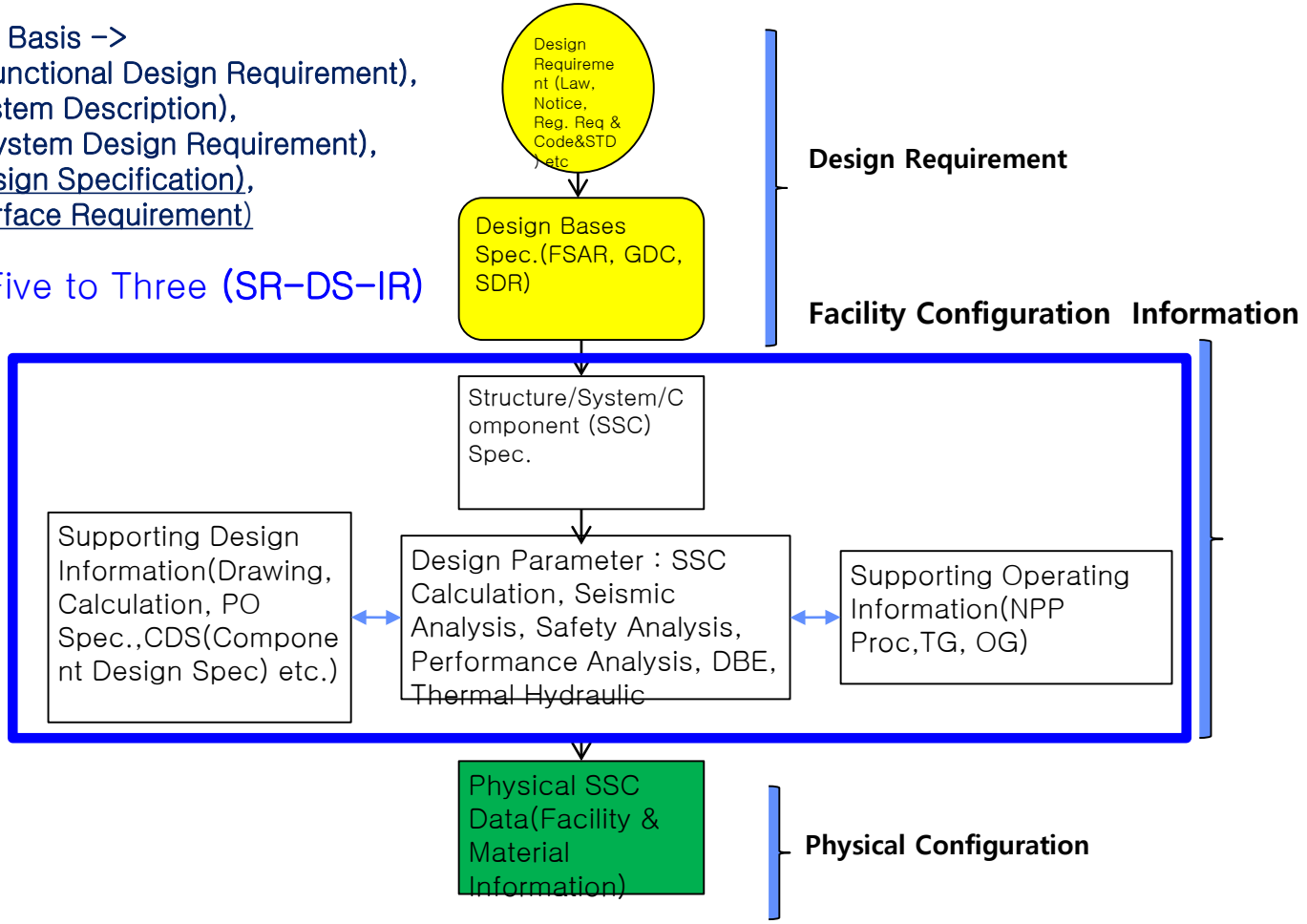
Requirements Definition Table

- Why create a requirements definition table?
- To manage requirements in a way that makes them identifiable, reviewable, and traceable.
- Requirements analysis tables can be used in structural analysis and design, object-oriented analysis and design, and component-based design (CBD).

[SSC] Hierarchical of Design Req.

Design Basis ->
FDR(Functional Design Requirement),
SD(System Description),
SDR(System Design Requirement),
DS(Design Specification),
IR(Interface Requirement)

Reducing Five to Three (SR-DS-IR)



Traceability Analysis from the Viewpoint of Development/Verification Methodologies (3)

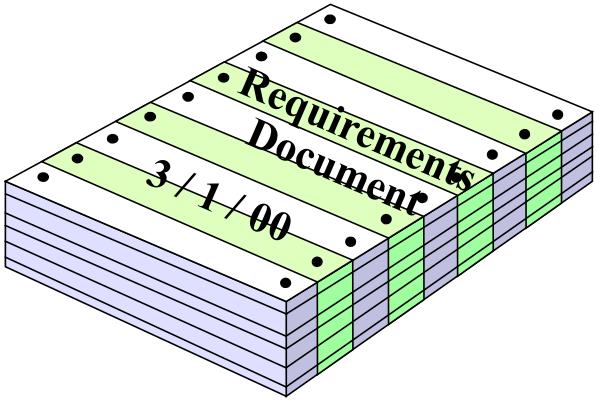
Category	Structural Analysis and Design	Object-Oriented Analysis and Design	CBD (Component-Based Development)	Applicability to Safety-Critical Software
Characteristics	Function- and procedure-centric decomposition	Object- and interaction-centric design	Combination of reusable components	
Design Perspective	Requirements → Functions → Modules	Requirements → Objects → Classes	Requirements → Components → Interfaces	
Traceability	Most clear and straightforward	Moderate; relationships may be complex	Interface-centric; internal traceability may be difficult	
Advantages	Simplicity, clarity, easy verification	Excellent maintainability and scalability	Excellent reusability and productivity	
Disadvantages	Low flexibility and reusability	Complex verification due to dynamic behavior	Opacity, high integration complexity	
App of Safety Critical SW	**Most advantageous**	Conditionally possible	Limited and conditionally possible	SA/SD> OOA/OOD> CBD

Requirements Engineering

- The process of establishing the services that the customer requires from a system and the constraints under which it operates and is developed
- Misunderstanding of the requirements, between the user and developer can result in major failures
- Criteria to determine implementation correctness
- It is NOT a design document. As far as possible, it should set of WHAT the system should do rather than HOW it should do it.
- Requirements may be functional or non-functional
  - Functional requirements describe system services or functions
  - Non-functional requirements is a constraint on the system or on the development process

How are Requirements Being Managed Today

- Most companies today manage requirements using :
  - Word processors/Spreadsheets
  - Sticky notes, phone calls, white boards
  - Email
  - Desktop databases



Document-Centric:Common Problems

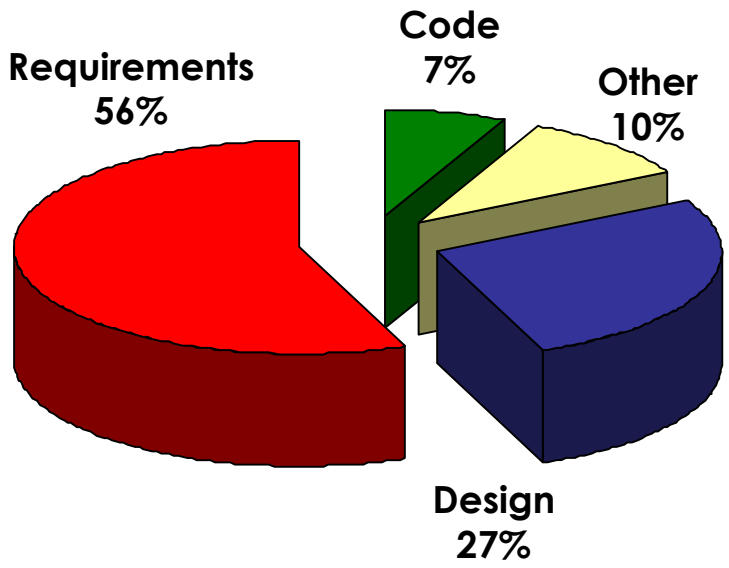
- Inability to collaborate between disparate locations
- Inability for simultaneous multiple users
- Lack of timely communication
- Wasted time maintaining documents
- Managing expectations (manage scope)
- Inability to analyze impact of changes
- No automated audit trail of changes
- Difficulty re-using requirements
- How do you know what to test?
- When is testing complete?

What Happens?

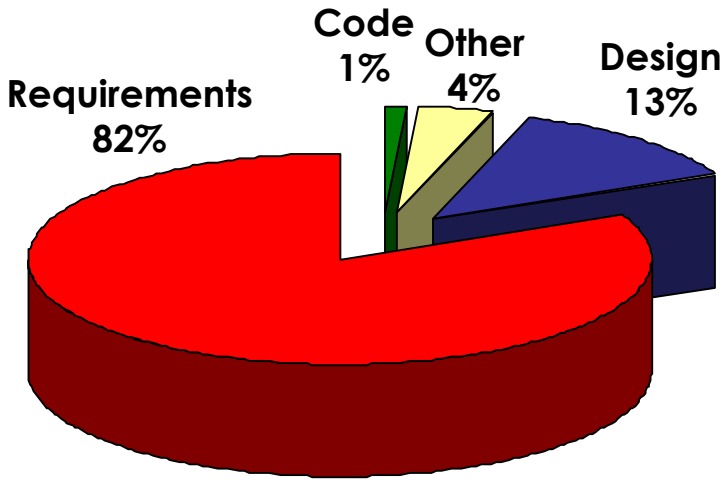
- Missed deadlines
- Cost overruns
- Unhappy end-users
- Costly defects
- IT’s reputation damaged

Why Manage Requirements ?

Distribution of Defects



Distribution of Effort to Fix Defects



Refer to: James Martin

Reasons for failure:

- The three reasons for failure and rework of requirements are:
  - ✓ ***Incomplete requirements and specifications***
  - ✓ ***Changing Requirements and Specifications***
  - ✓ ***Lack of user input***

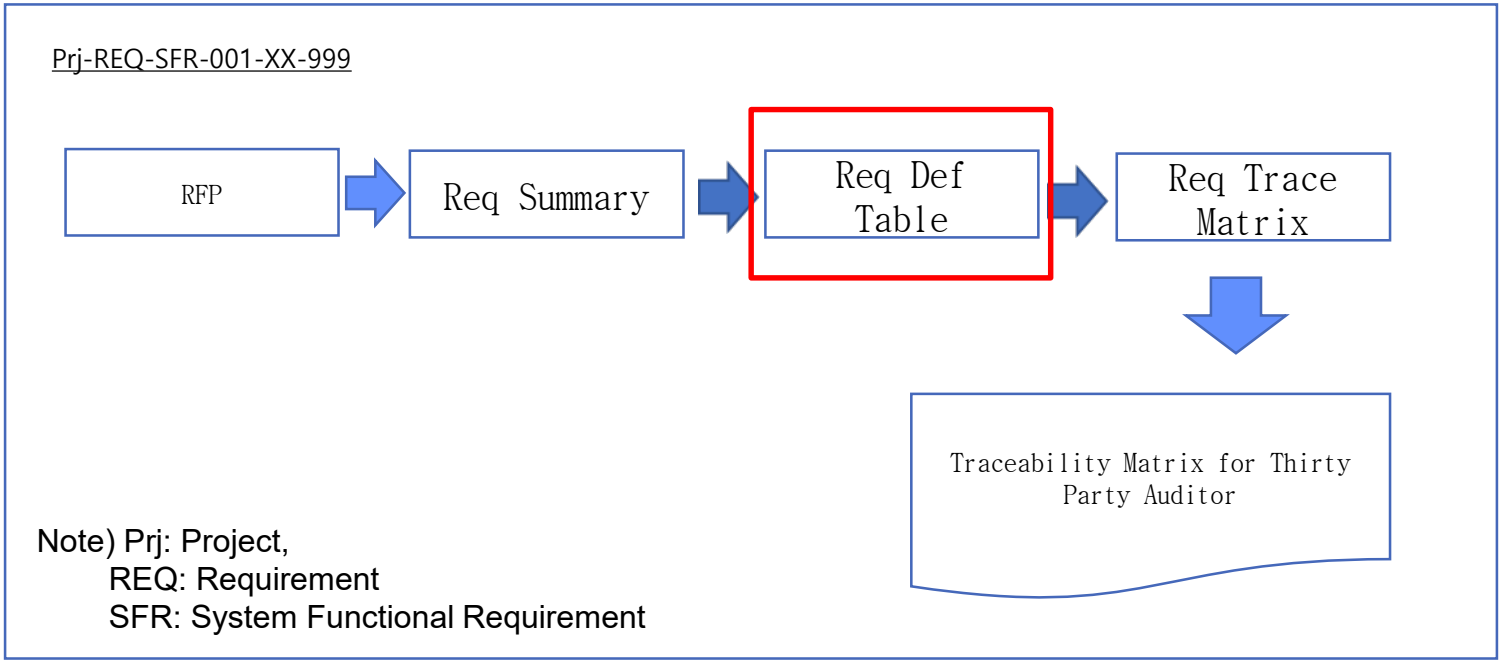
Cost-Effective approach for Requirement Specifications

	Sil 4	Sil 3	Sil 2	Sil 1
Development Methods	Formal method	Semi-formal method	Semi-formal method	Informal method
Application Domain	Plant protection system	Control system	Monitoring system Operation support system	Others
Formality	Z, VDM, Larch, CSP, CCS. Statechart, Temporal Logic, I/O automata RAISE, LOTUS, SCR	SA/SD Semi-graphic representation Approach of Chen, Yourdon-DeMacro, Hatly-Pirbhai, Yourdon-Constantine, Booch-Buhr etc.	Same as Sil 2	Coding & Fixing

Requirements Summary(Assuming Component Based Modeling)

Type of Req	Definition	Abbreviation	Comments
{Consulting Req}	(If necessary)	CSR	< Requirement ID> REQ-SFR-001-XX-999 ---- ---------- ---- ---- ① ② ③ ④ ① REQ: Requirement Number Classification ② SFR-001: RFP Requirement ID New Assignment: SFR-000 ③ XX: Sub_Work ID Classification (2-letter uppercase English letters) ④ 999: Serial Number (001~999, assigned per task) ex) Functional Safety: REQ-SFR-001-FS-123 Timing Management: REQ-SFR-085-TM-567 In case of missing requirement -> (for example) REQ-SFR-001-FS-123-**01**, REQ-SFR-001-FS-123-**02**
System Function Req	Target System, Function(Operational Scenario)	SFR	
Performance Req	A description of the target system's performance, including processing speed and time, throughput, dynamic and static capacity, and availability.	PER	
Sys I/F Req		INT	
Data Req	Database design and data establishment etc.	DAR	
Maintenance Req		MAR	
Test Req		TER	
Security Req		SER	
Quality Req		QUR	
Constraint Req	Functional, non-functional, interface, and data requirements, as well as constraints necessary for building the system	COR	
Project Mg't Req		PMR	
Project Support Req		PSR	

Process of Req Tracing



RFP Template

Requirement ID			
Requirement Name			
Requirement Class		Acceptance Level	
Requirement Description			
Output Information			

Requirement Categories and Identifier

Requirement Category	Identifier	Requirement Definition	Number of Requirements
Function Requirement	SFR-Serial No.	SFR-Sub Req	
Performance Requirement	PER-Serial No.	PER--Sub Req	
Interface	SIR-Serial No.	SIR-Sub Req	
Data	DAR-Serial No.	DAR-Sub Req	
Test	TER-Serial No.	TER-Sub Req	
Security	SER-Serial No.	SER-Sub Req	
Quality	QUR-Serial No.	QUR-Sub Req	
Constraint	COR-Serial No.	COR-Sub Req	
Project Management	PMR-Serial No.	PMR-Sub Req	

Requirements Definition Table

RFP	Req. Type	Req ID	Addition or Deletion	Req Name	Request/Change History	Design Output	Accept (Y/N)	Functional/Non-Functional	Source		Notes	Audit Review
									(RFP)	Basis		

When creating the Requirements Definition Table, requirements IDs must be assigned uniquely as follows.

Prj-REQ-SFR-001-XX-999

① ② ③ ④ ⑤

① Project Name

② REQ: Requirement Identifier

③ SFR-001: RFP Requirement ID

④ XX: Detailed Task Classification (2 or 3 uppercase letters)

⑤ 999: Serial Number (001~999, assigned per task); when adding omitted numbers, append a lowercase English letter after 999 (e.g., 999a, 999b, etc.)

- If missing req, go back to initial [by CBM only]
- If case of expansion, needed to be additional field -01, -02, 03

Requirements Traceability Matrix

No.	RFP	No.	Req ID	Added Yes/No	Works ID	Req Name (RFP)	Req Details/Change History	Implementation Plan	Acceptance (Yes/No)	Type	Infor Source	Source Evidence

The Person in Charge	Process ID	Window ID	Program ID	Interface ID	Unit Test ID	Integration Test ID	Inspection Item	Inspection Criteria	Inspection t Pass/Fail	Remarks

Building a Traceability Matrix for Auditors

No	RFP	Task ID	Req ID	Req Definition Table	CM-No	Traceability Analysis Table	Architect Design Document	Process ID	Window ID	Window Design/Menu Structure	I/F Window	Interface

Notes: CM: Configuration Management

Conclusion

- We can create a **requirements definition table** based on the RFP. visibility, and facilitating communication
- The Requirements Definition Table can be used regardless of the development methodology. However, the subsequent stages following the requirements phase may vary depending on the chosen methodology.
- In traditional structural analysis and design, forward, backward, and bidirectional approaches are essential.
- In case of Component Based Modelling can be applied for all-in-one approach