

# Software Architecture Documentation

Integrating Rozanski & Woods with Views and Beyond

A Comprehensive Guide to Mapping Viewpoints,  
Perspectives, and Documentation Practices

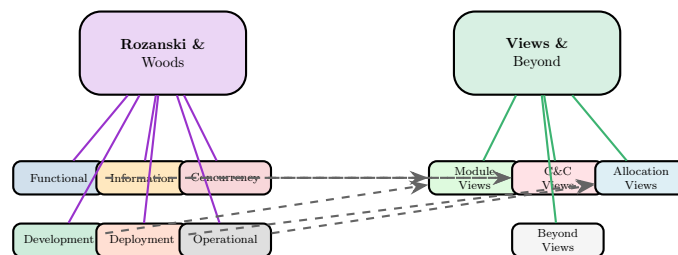
*Architecture Documentation Series*

Based on Software Systems Architecture (R&W) and SEI Views and Beyond

December 9, 2025

## Abstract

Rozanski and Woods' "Software Systems Architecture" and the SEI's "Views and Beyond" represent two of the most influential approaches to software architecture documentation. While both frameworks share common foundations in IEEE 1471/ISO 42010, they offer distinct perspectives and terminology. This comprehensive guide provides detailed mappings between the six Rozanski & Woods viewpoints and Views and Beyond view types, addresses the R&W perspectives and their V&B equivalents, and offers practical guidance for creating documentation that leverages the strengths of both approaches. Whether you are familiar with one framework and need to work with the other, or seeking to create documentation that satisfies multiple stakeholder communities, this guide provides the integration knowledge you need.



Viewpoint Integration

## Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	Purpose of This Guide . . . . .	3
1.2	Framework Comparison . . . . .	3
1.3	Core Conceptual Mappings . . . . .	3
<b>2</b>	<b>Framework Overviews</b>	<b>4</b>
2.1	Rozanski & Woods Viewpoints . . . . .	4
2.2	Rozanski & Woods Perspectives . . . . .	5
2.3	Views and Beyond Structure . . . . .	5
<b>3</b>	<b>Functional Viewpoint Mapping</b>	<b>6</b>
<b>4</b>	<b>Information Viewpoint Mapping</b>	<b>7</b>
<b>5</b>	<b>Concurrency Viewpoint Mapping</b>	<b>8</b>
<b>6</b>	<b>Development Viewpoint Mapping</b>	<b>9</b>
<b>7</b>	<b>Deployment Viewpoint Mapping</b>	<b>10</b>
<b>8</b>	<b>Operational Viewpoint Mapping</b>	<b>11</b>
<b>9</b>	<b>Perspectives and Quality Attributes</b>	<b>12</b>
9.1	Mapping Perspectives to V&B Approaches . . . . .	12
9.2	Security Perspective . . . . .	13
9.3	Performance and Scalability Perspective . . . . .	13
9.4	Availability and Resilience Perspective . . . . .	14
<b>10</b>	<b>Complete Mapping Reference</b>	<b>14</b>
<b>11</b>	<b>Implementation Guidance</b>	<b>15</b>
11.1	Creating Integrated Documentation . . . . .	16
11.2	Documentation Structure Options . . . . .	16
11.3	View Packet Design . . . . .	16
<b>12</b>	<b>Appendix A: Quick Reference Mapping</b>	<b>17</b>
<b>13</b>	<b>Appendix B: Glossary</b>	<b>18</b>
<b>14</b>	<b>Appendix C: References</b>	<b>18</b>

# 1 Introduction

## 1.1 Purpose of This Guide

This guide provides comprehensive integration between Rozanski and Woods’ viewpoint catalog (from “Software Systems Architecture”) and the SEI’s Views and Beyond approach. Both frameworks are widely used and influential, and practitioners often need to:

- Translate between frameworks when working with different teams
- Create documentation that satisfies stakeholders familiar with either approach
- Leverage the unique strengths of each framework
- Understand the conceptual relationships between the approaches

## 1.2 Framework Comparison

### Definition

**Rozanski & Woods (R&W):** A practical approach to software systems architecture that defines six core viewpoints and multiple perspectives. Emphasizes stakeholder concerns and provides detailed viewpoint catalogs with concerns, models, and problems/pitfalls.

### Definition

**Views and Beyond (V&B):** An SEI approach that organizes architecture documentation into three view categories (Module, C&C, Allocation) plus documentation beyond views. Provides detailed guidance for documenting views and view packets.

Table 1: Framework Comparison

Aspect	Rozanski & Woods	Views and Beyond
Origin	Industry practice (UK)	SEI research (US)
Standard	IEEE 1471 / ISO 42010	IEEE 1471 / ISO 42010
Alignment		
View	6 viewpoints	3 view categories
Organization		
Cross-Cutting	Perspectives	Quality attribute analysis
Documentation	Viewpoint catalog	View packet
Unit		
Stakeholder	Explicit concern mapping	Stakeholder-driven view selection
Focus		
Quality	Perspectives	Scenarios and tactics
Attributes		

## 1.3 Core Conceptual Mappings

Table 2: Conceptual Term Mappings

R&W Term	V&B Term	Notes
Viewpoint	View Type / Style	Conventions for view construction
View	View	Work product for specific concerns
Model	Model	Representation within a view
Perspective	Quality Attribute Analysis	Cross-cutting concerns
Concern	Concern	Stakeholder interest
Catalog Entry	View Packet	Documented view content
Element	Element	Architectural building block

## 2 Framework Overviews

### 2.1 Rozanski & Woods Viewpoints

R&W defines six core viewpoints, each addressing specific stakeholder concerns:

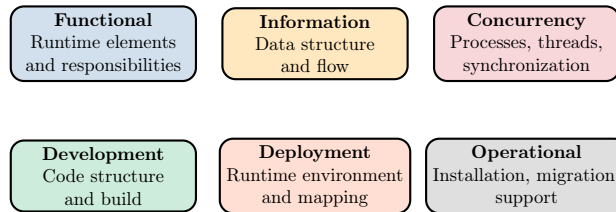


Figure 1: Rozanski &amp; Woods Six Viewpoints

Table 3: R&amp;W Viewpoints Summary

Viewpoint	Primary Concerns	Typical Models
Functional	Functional capabilities; external interfaces; internal structure	Functional structure model; Component diagram
Information	Data structure; data flow; data ownership; data quality	Static data structure; Data flow; Data lifecycle
Concurrency	Task structure; process mapping; synchronization	Concurrency model; State model; Process diagram
Development	Module organization; codeline; common design	Module structure; Codeline model; Common design

Deployment	Runtime platform; network; technology dependencies	Deployment model; Network model; Technology model
Operational	Installation; migration; configuration; support	Installation model; Migration model; Config model

## 2.2 Rozanski & Woods Perspectives

R&W perspectives address cross-cutting quality concerns:

Table 4: R&W Perspectives

Perspective	Concerns	Applicable Viewpoints
Security	Authentication; authorization; confidentiality; integrity	All viewpoints
Performance & Scalability	Response time; throughput; scalability	Functional, Concurrency, Deployment
Availability & Resilience	Availability; reliability; recovery; resilience	Functional, Concurrency, Deployment
Evolution	Dimensions of change; magnitude; likelihood	Development, Deployment
Location	Geographic distribution; data location	Deployment, Information
Development Resource	Team skills; schedule; budget	Development
Internationalization	Language; locale; cultural	Information, Functional
Accessibility	Disability access; usability	Functional
Regulation	Compliance; audit; legal	All viewpoints
Usability	Ease of use; learnability	Functional

## 2.3 Views and Beyond Structure

V&B organizes views into three categories:

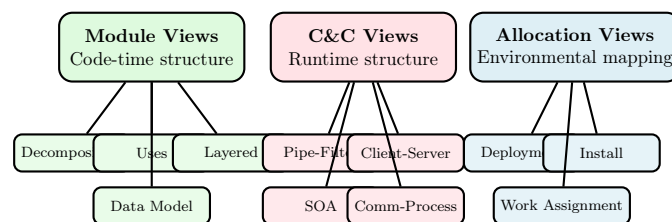


Figure 2: Views and Beyond Structure

### 3 Functional Viewpoint Mapping

#### R&W Functional Viewpoint

**Purpose:** Describes the system's runtime functional elements and their responsibilities, interfaces, and primary interactions.

**Concerns Addressed:**

- Functional capabilities
- External interfaces
- Internal structure
- Functional design philosophy

**Models:**

- Functional Structure Model
- External Interface Model
- Component diagrams

**Stakeholders:** All stakeholders; particularly users, acquirers, developers, testers

#### Views and Beyond Equivalent

**Primary Mapping:** Component-and-Connector (C&C) View Styles

The Functional viewpoint maps to one or more C&C styles in V&B:

- **Client-Server Style:** For request-response interactions
- **Service-Oriented Style:** For service-based architectures
- **Pipe-and-Filter Style:** For data transformation pipelines
- **Publish-Subscribe Style:** For event-driven systems
- **Shared-Data Style:** For repository-centered systems

**Why C&C:** Both the Functional viewpoint and C&C views focus on runtime structure, component responsibilities, and interactions.

#### Implementation Mapping for Functional Viewpoint

**R&W Functional Structure Model → V&B C&C Primary Presentation**

- Functional elements → Components
- Interfaces → Ports
- Interactions → Connectors
- Responsibilities → Element catalog properties

**R&W External Interface Model → V&B Context Diagram + Interface Documentation**

- External entities → Context diagram actors
- Interface specifications → Interface documentation template

**R&W Concerns → V&B Documentation**

- Functional capabilities → Element catalog responsibilities
- Design philosophy → Rationale documentation

**Example****Functional View Mapping Example****R&W Functional Element:**

“OrderProcessor: Receives customer orders, validates them, calculates pricing, and submits to fulfillment.”

**V&B C&C Element Catalog Entry:**

*Element:* OrderProcessor (Component)

*Responsibilities:* Order validation; Pricing calculation; Fulfillment submission

*Interfaces:*

- OrderSubmission (provided): Accepts customer orders
  - FulfillmentRequest (required): Submits to fulfillment system
- Quality Attributes:* Process 100 orders/second; 99.9% availability

## 4 Information Viewpoint Mapping

**R&W Information Viewpoint**

**Purpose:** Describes the way the system stores, manipulates, manages, and distributes information.

**Concerns Addressed:**

- Data structure and content
- Data flow
- Data ownership and management
- Data quality (timeliness, availability, accuracy)
- Data volumes

**Models:**

- Static Data Structure Model (entities and relationships)
- Data Flow Model
- Data Ownership Model
- Data Lifecycle Model

**Stakeholders:** Users, developers, DBAs, compliance officers

**Views and Beyond Equivalent**

**Primary Mapping:** Data Model Style (Module Views) + C&C Styles for Flow

- **Data Model Style:** For static data structure (entities, relationships, attributes)
- **Shared-Data C&C Style:** For data stores and accessor components
- **Pipe-and-Filter C&C Style:** For data flow and transformations
- **Repository C&C Style:** For centralized data management

**Additional V&B Support:**

- Interface documentation for data contracts
- Quality attribute scenarios for data quality

- Rationale for data ownership decisions

### Implementation Mapping for Information Viewpoint

#### R&W Static Data Structure → V&B Data Model View

- Entities → Data entities in data model
- Relationships → Relations in data model
- Attributes → Entity properties

#### R&W Data Flow Model → V&B C&C View (Pipe-Filter or Shared-Data)

- Data flows → Connectors with data semantics
- Transformation points → Filter components
- Data stores → Repository components

#### R&W Data Ownership → V&B Element Catalog + Rationale

- Ownership assignments → Element properties
- Ownership rationale → Rationale documentation

#### R&W Data Quality Concerns → V&B Quality Attribute Scenarios

- Timeliness → Performance scenarios
- Availability → Availability scenarios
- Accuracy → Integrity requirements

## 5 Concurrency Viewpoint Mapping

### R&W Concurrency Viewpoint

**Purpose:** Describes the concurrency structure of the system and maps functional elements to concurrency units.

#### Concerns Addressed:

- Task structure
- Mapping of functional elements to tasks
- Interprocess communication
- State management
- Synchronization and integrity
- Startup, shutdown, and failure behavior

#### Models:

- System-level Concurrency Model
- State Model
- Process Structure Diagram

**Stakeholders:** Developers, testers, performance engineers, operations

### Views and Beyond Equivalent

**Primary Mapping:** Communicating-Processes C&C Style

The V&B Communicating-Processes style directly addresses concurrency:

- **Elements:** Processes, threads, or concurrent units
- **Connectors:** Communication mechanisms (synchronous, asynchronous, shared memory)



- **Properties:** Thread safety, synchronization, scheduling

**Additional V&B Support:**

- Behavior documentation for state transitions
- Sequence diagrams for interaction scenarios
- Quality attribute scenarios for performance

### Implementation Mapping for Concurrency Viewpoint

**R&W Concurrency Model → V&B Communicating-Processes C&C View**

- Tasks/Processes → Process components
- Interprocess communication → Connector types
- Synchronization → Connector properties

**R&W State Model → V&B Behavior Documentation**

- State diagrams → Behavior documentation within C&C view
- State transitions → State machine notation

**R&W Task Mapping → V&B View Mapping**

- Functional-to-task mapping → Mapping between C&C views

## 6 Development Viewpoint Mapping

### R&W Development Viewpoint

**Purpose:** Describes the architecture that supports the software development process.

**Concerns Addressed:**

- Module organization
- Common processing (shared code)
- Standardization of design
- Standardization of testing
- Codeline organization
- Build and configuration management

**Models:**

- Module Structure Model
- Common Design Model
- Codeline Model

**Stakeholders:** Developers, testers, build engineers, maintainers

### Views and Beyond Equivalent

**Primary Mapping:** Module View Styles + Documentation Beyond Views

- **Decomposition Style:** For module structure (hierarchy of modules)
- **Layered Style:** For layer structure and dependencies
- **Uses Style:** For module dependencies and allowed usage
- **Implementation/Install Style:** For codeline model

**Additional V&B Support:**

- Rationale for common design decisions
- Documentation beyond views for conventions
- Work Assignment view for team structure

**Implementation Mapping for Development Viewpoint****R&W Module Structure Model → V&B Decomposition/Layered View**

- Modules → Modules in decomposition
- Layers → Layers in layered view
- Dependencies → Uses relations

**R&W Common Design Model → V&B Documentation Beyond Views**

- Common patterns → Design guidance in rationale
- Design conventions → Documented in system overview
- Shared assumptions → Documented assumptions

**R&W Codeline Model → V&B Implementation/Install Style**

- Source structure → Implementation view
- Build organization → Install view
- Configuration management → Documentation beyond views

## 7 Deployment Viewpoint Mapping

**R&W Deployment Viewpoint**

**Purpose:** Describes the environment into which the system will be deployed and the dependencies the system has on its runtime environment.

**Concerns Addressed:**

- Runtime platform requirements
- Hardware requirements
- Network requirements
- Technology compatibility
- Network capacity

**Models:**

- Runtime Platform Model
- Network Model
- Technology Dependency Model

**Stakeholders:** Operations, developers, system administrators, acquirers

**Views and Beyond Equivalent**

**Primary Mapping:** Deployment Style (Allocation Views) + Uses Style

- **Deployment Style:** For runtime platform and network models
  - Software elements mapped to hardware nodes
  - Network topology
  - Resource allocation

- **Uses Style:** For technology dependency model
  - Dependencies on runtime environment
  - Library and framework dependencies

### Implementation Mapping for Deployment Viewpoint

#### R&W Runtime Platform Model → V&B Deployment View

- Processing nodes → Deployment nodes
- Software-to-hardware mapping → Allocation relations
- Resource requirements → Node properties

#### R&W Network Model → V&B Deployment View

- Network topology → Node connections
- Network properties → Connection properties
- Network segments → Environmental regions

#### R&W Technology Dependency Model → V&B Uses View + Element Catalog

- Technology dependencies → Uses relations
- Version requirements → Element catalog properties
- Compatibility constraints → Variability guide

## 8 Operational Viewpoint Mapping

### R&W Operational Viewpoint

**Purpose:** Describes how the system will be operated, administered, and supported when running in production.

#### Concerns Addressed:

- Installation and upgrade
- Functional migration
- Data migration
- Operational monitoring and control
- Configuration management
- Support and maintenance

#### Models:

- Installation Model
- Migration Model
- Configuration Model
- Administration Model
- Support Model

**Stakeholders:** Operations, system administrators, support staff, maintainers

### Views and Beyond Equivalent

**Primary Mapping:** Install Style + Documentation Beyond Views

- **Install Style:** For installation model
  - File system structure

- Configuration locations
- Installation procedures
- **Documentation Beyond Views:** For operational requirements and solutions
  - Operational procedures in system overview
  - Migration plans in rationale
  - Configuration guidance

Operational concerns can be associated with any view where solutions are documented.

### Implementation Mapping for Operational Viewpoint

#### R&W Installation Model → V&B Install View

- Installation structure → Install view primary presentation
- Installation procedures → Associated documentation

#### R&W Migration Model → V&B Documentation Beyond Views

- Migration strategy → Rationale / system overview
- Migration procedures → Operational documentation

#### R&W Configuration Model → V&B Variability Guide + Install View

- Configuration options → Variability guide
- Configuration locations → Install view

#### R&W Operations Concerns → V&B View Annotations

- Monitoring requirements → C&C view properties
- Administration requirements → Deployment view properties
- Support requirements → Documentation beyond views

## 9 Perspectives and Quality Attributes

### 9.1 Mapping Perspectives to V&B Approaches

R&W perspectives are cross-cutting concerns that apply across viewpoints. V&B addresses these through quality attribute analysis.

Table 5: Perspective to V&B Mapping

R&W Perspective	V&B Approach	Documentation Location
Security	Security tactics; Security view	C&C view; Deployment view; Rationale
Performance & Scalability	Performance scenarios; Performance tactics	C&C view properties; Rationale
Availability & Resilience	Availability scenarios; Tactics	C&C view; Deployment view; Behavior
Evolution	Modifiability scenarios; Variability	Variability guide; Rationale
Location	Deployment allocation	Deployment view
Development	Work assignment	Work Assignment view
Resource		

R&W Perspective	V&B Approach	Documentation Location
Internationalization	Data model; Configuration	Data model view; Variability guide
Accessibility	Functional requirements	C&C view; Interface documentation
Regulation	Constraints; Compliance	Rationale; Constraints documentation
Usability	Functional requirements	C&C view; Interface documentation

## 9.2 Security Perspective

### Security Perspective Mapping

#### R&W Security Concerns:

- Authentication and authorization
- Confidentiality and integrity
- Non-repudiation and audit
- Threat identification

#### V&B Documentation Approach:

- **C&C View:** Security components (authentication services, encryption services)
- **Deployment View:** Security zones, network segmentation, firewall placement
- **Rationale:** Security decisions and threat analysis
- **Quality Scenarios:** Security-specific scenarios (e.g., “Attacker attempts SQL injection; system detects and blocks within 1 second”)

## 9.3 Performance and Scalability Perspective

### Performance & Scalability Perspective Mapping

#### R&W Performance Concerns:

- Response time and throughput
- Scalability dimensions
- Resource utilization
- Peak load handling

#### V&B Documentation Approach:

- **C&C View:** Performance-critical components; Connector properties (latency, bandwidth)
- **Deployment View:** Resource allocation; Scaling units
- **Element Catalog:** Performance properties per element
- **Quality Scenarios:** Performance scenarios with specific measures
- **Rationale:** Performance tactics applied and tradeoffs made

## 9.4 Availability and Resilience Perspective

### Availability & Resilience Perspective Mapping

#### R&W Availability Concerns:

- Availability targets
- Failure detection and recovery
- Redundancy and failover
- Disaster recovery

#### V&B Documentation Approach:

- **C&C View:** Redundant components; Health monitoring; Circuit breakers
- **Deployment View:** Redundancy placement; Failover paths; Disaster recovery sites
- **Behavior Documentation:** Failure scenarios; Recovery procedures
- **Quality Scenarios:** Availability scenarios (e.g., “Database fails; system recovers within 30 seconds”)

## 10 Complete Mapping Reference

Table 6: Complete Viewpoint Mapping Reference

R&W Viewpoint	V&B View Type(s)	Implementation Notes
<b>Functional Viewpoint</b>		
Functional Structure	C&C Views (Client-Server, SOA, Pipe-Filter)	Choose style based on interaction patterns
External Interfaces	Context Diagram + Interface Documentation	Use V&B interface template
Design Philosophy	Rationale Documentation	Document in system overview
<b>Information Viewpoint</b>		
Static Data Structure	Data Model View	Entity-relationship notation
Data Flow	C&C View (Pipe-Filter, Shared-Data)	Document data semantics on connectors
Data Ownership	Element Catalog Properties	Document in data model view
Data Quality	Quality Attribute Scenarios	Create data-specific scenarios
<b>Concurrency Viewpoint</b>		
Concurrency Model	C&C View (Communicating-Processes)	Document processes, synchronization

<b>R&amp;W Viewpoint</b>	<b>V&amp;B View Type(s)</b>	<b>Implementation Notes</b>
State Model	Behavior Documentation	State diagrams within C&C view
Task Mapping	View Mapping	Map functional to concurrency views
<b>Development Viewpoint</b>		
Module Structure	Decomposition View, Layered View	Use for code organization
Common Design	Documentation Beyond Views	Document patterns, conventions
Codeline Model	Implementation/Install View	Source structure, build
<b>Deployment Viewpoint</b>		
Runtime Platform	Deployment View	Software-to-hardware mapping
Network Model	Deployment View	Node connections, topology
Technology Dependencies	Uses View	Dependencies on environment
<b>Operational Viewpoint</b>		
Installation Model	Install View	File structure, config locations
Migration Model	Documentation Beyond Views	Migration strategy in rationale
Configuration Model	Variability Guide	Configuration options
Operations Concerns	View Annotations	Annotate relevant views

## 11 Implementation Guidance

## 11.1 Creating Integrated Documentation

### Best Practice

#### Integration Strategy:

1. **Start with stakeholder analysis:** Identify stakeholders familiar with each framework
2. **Map required content:** Determine which R&W viewpoints are needed and map to V&B views
3. **Choose primary organization:** Decide whether to organize by R&W viewpoints or V&B categories
4. **Create cross-references:** Provide navigation aids for stakeholders familiar with either framework
5. **Document perspectives/quality attributes:** Ensure cross-cutting concerns are addressed in appropriate views

## 11.2 Documentation Structure Options

Table 7: Documentation Organization Options

Approach	Organization	When to Use
R&W Primary	Organize by R&W viewpoints; note V&B equivalents	Stakeholders familiar with R&W; need viewpoint structure
V&B Primary	Organize by V&B categories; note R&W coverage	Stakeholders familiar with V&B; SEI-aligned projects
Hybrid	Use V&B structure with R&W viewpoint mapping	Mixed stakeholder community; need both frameworks
Concern-Driven	Organize by stakeholder concerns; map to both	Custom organization; highly diverse stakeholders

## 11.3 View Packet Design

When creating view packets that satisfy both frameworks:



**View Packet Completeness Checklist****For V&B Completeness:**

- ☐ Primary presentation (diagram)
- ☐ Element catalog
- ☐ Context diagram (if applicable)
- ☐ Variability guide
- ☐ Rationale
- ☐ Behavior documentation (if applicable)

**For R&W Viewpoint Coverage:**

- ☐ All viewpoint concerns addressed
- ☐ Required models included
- ☐ Stakeholder needs met
- ☐ Problems/pitfalls considered

**For Perspective/Quality Attribute Coverage:**

- ☐ Cross-cutting concerns documented
- ☐ Quality scenarios defined
- ☐ Tactics documented in rationale

## 12 Appendix A: Quick Reference Mapping

Table 8: Quick Reference: R&amp;W to V&amp;B Mapping

R&W View/Model	V&B Approach
Functional View	One or more C&C styles
Information View (Structure)	Data Model style
Information View (Flow)	C&C style (Pipe-Filter, Shared-Data)
Concurrency View	C&C Communicating-Processes style
Development View (Structure)	Decomposition or Layered style
Development View (Codeline)	Implementation/Install style
Development View (Common Design)	Documentation beyond views
Deployment View (Platform)	Deployment style
Deployment View (Network)	Deployment style
Deployment View (Technology)	Uses style

Operational View (Installation)	Install style
Operational View (Operations)	Documentation beyond views
Security Perspective	Security annotations in C&C/Deployment; Rationale
Performance Perspective	Performance properties; Quality scenarios; Rationale
Availability Perspective	Availability tactics in views; Quality scenarios
Evolution Perspective	Variability guide; Modifiability scenarios

## 13 Appendix B: Glossary

<b>C&amp;C View</b>	Component-and-Connector view; shows runtime structure
<b>Concern</b>	Stakeholder interest in the system
<b>Element Catalog</b>	Detailed specification of view elements
<b>Module View</b>	View showing code-time structure
<b>Perspective</b>	R&W cross-cutting quality concern
<b>Quality Attribute</b>	Measurable system property (performance, security, etc.)
<b>Style</b>	V&B term for a type of view with specific element/relation types
<b>View</b>	Work product expressing architecture for specific concerns
<b>View Packet</b>	V&B unit of view documentation
<b>Viewpoint</b>	Conventions for constructing and interpreting views

## 14 Appendix C: References

1. Rozanski, N., & Woods, E. (2011). *Software Systems Architecture: Working with Stakeholders Using Viewpoints and Perspectives* (2nd ed.). Addison-Wesley.
2. Clements, P., et al. (2010). *Documenting Software Architectures: Views and Beyond* (2nd ed.). Addison-Wesley.
3. Bass, L., Clements, P., & Kazman, R. (2021). *Software Architecture in Practice* (4th ed.). Addison-Wesley.
4. ISO/IEC/IEEE 42010:2011. *Systems and software engineering—Architecture description*.
5. Kruchten, P. (1995). "The 4+1 View Model of Architecture." *IEEE Software*, 12(6), 42-50.
6. IEEE 1471-2000. *IEEE Recommended Practice for Architectural Description of Software-Intensive Systems*.
7. Hofmeister, C., Nord, R., & Soni, D. (2000). *Applied Software Architecture*. Addison-Wesley.

8. Garlan, D. (2003). “Formal Modeling and Analysis of Software Architecture.” *Formal Methods for Software Architectures*, LNCS 2804.