

ARCHITECTURAL DESCRIPTION



Cristiane Lana e Délcio Silva

Software Architecture - SSC9441

Prof. Dr. Elisa Yumi Nakagawa

AGENDA

- Introduction
- Architectural Description
- Understand the ISO/IEC/IEEE 42010:2011
- Architectural Frameworks
- Final considerations
- Interesting Links

INTRODUCTION

- To build a software architecture is necessary to **know the stakeholders**
- Architectural description is a **artifact tangible**
- Help in the **communication process with stakeholders**
- Implementation **assessment**
- Entry for tool
 - Example.: **simulation**



INTRODUCTION

- It provides a common language to express different interests.

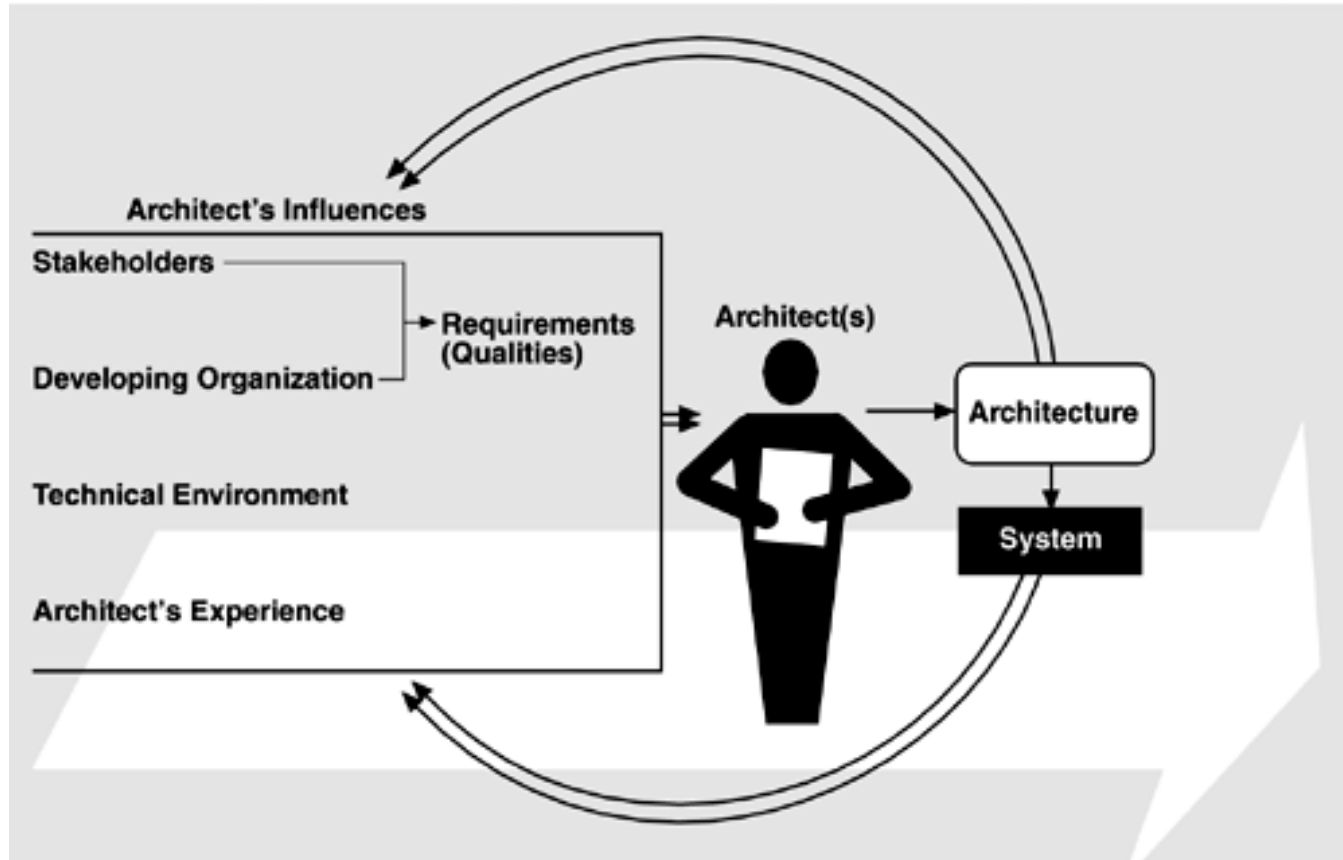


Copyright 1999-2005 Bredemeyer Consulting
<http://www.bredemeyer.com>

IndyJUG April 2005
WhatArchitectureDefinition.ppt Slide 15

INTRODUCTION

- It can be influenced by:



ARCHITECTURAL DESCRIPTION (AD)

- What is AD?
 - It is a **artifact** that express a **system architecture**
 - It describe a possible **architecture** for the system
 - It can be:
 - Document, model groups, model repository, among others

[ISO/IEC/IEEE 42010]

ARCHITECTURAL DESCRIPTION (AD)

- Why it is made:
- Support to:
 - Understand
 - Analyze
 - compare Architectures
 - Used as “blueprints” for planning and construction
 - Documentation
 -

[ISO/IEC/IEEE 42010]

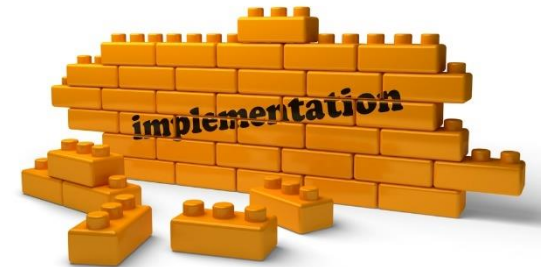
ARCHITECTURAL DESCRIPTION (AD)

- **How is it done?:**
 - It can be used as templates that support the description process
 - It is dependent of the *stakeholders*
 - Based on *interests and domain*
 - Choose the view that make up the system architecture

[ISO/IEC/IEEE 42010]

ARCHITECTURAL DESCRIPTION (AD)

- When is it done?



www.artsbma.org www.batalas.co.uk [ISO/IEC/IEEE 42010]

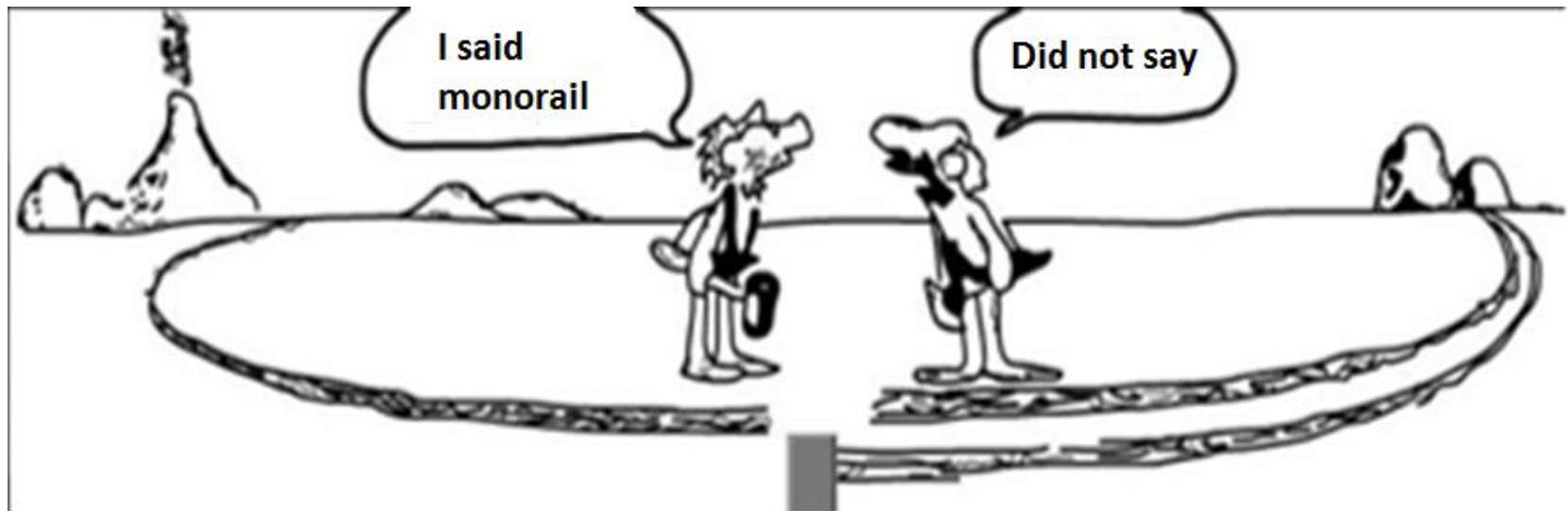
ARCHITECTURAL DESCRIPTION (AD)

- Problems with communication in the requirements phase:



ARCHITECTURAL DESCRIPTION (AD)

- When it is not done correctly:



ARCHITECTURAL DESCRIPTION (AD)

- **For who?**



and



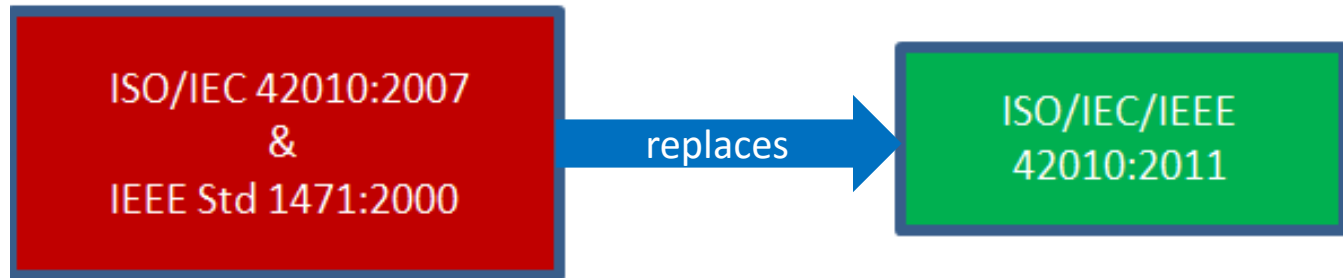
<http://www.crisismanagement.tony-Ridley.com>

ARCHITECTURAL DESCRIPTION (AD)

- Who does it?

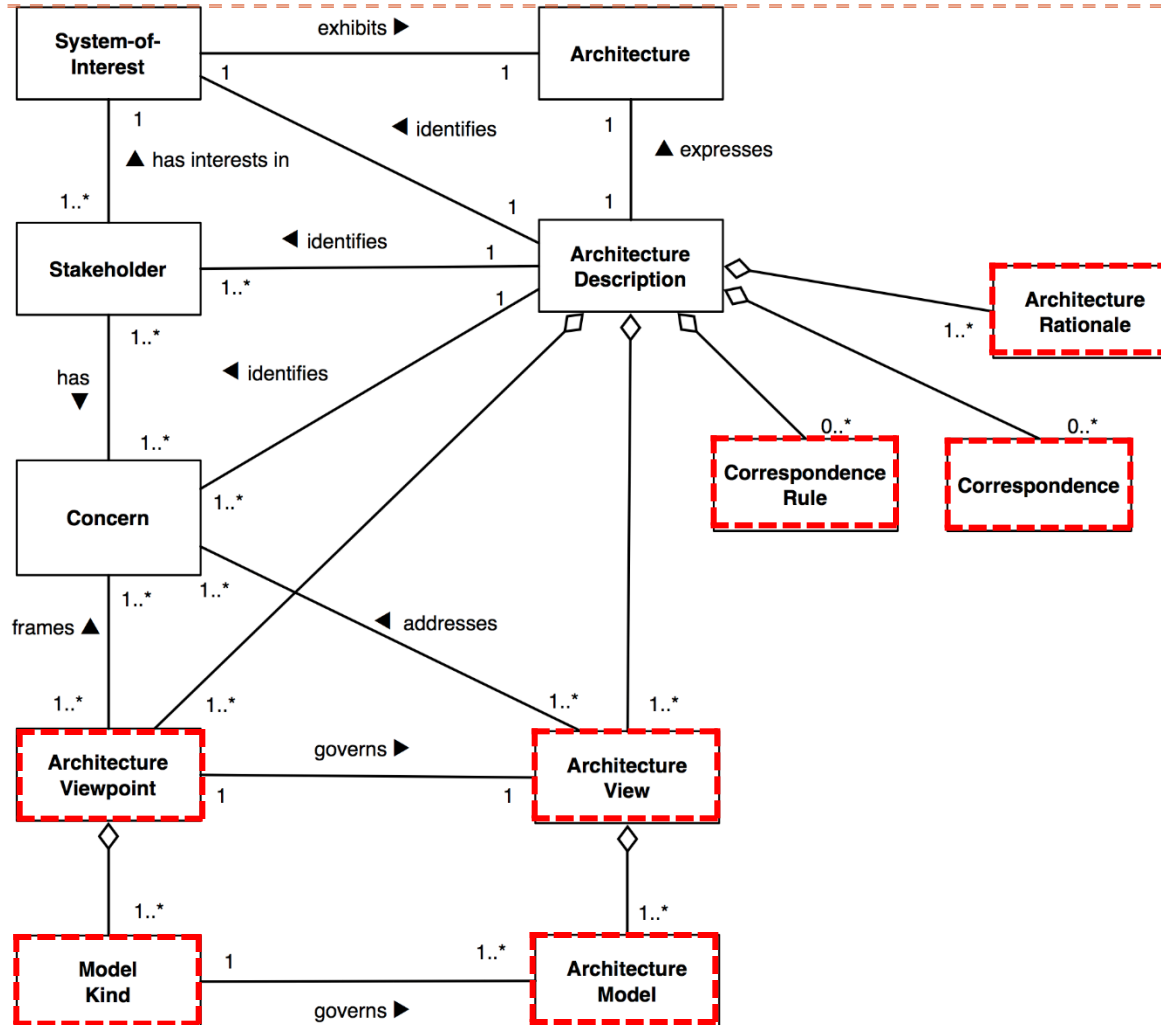


UNDERSTAND THE ISO/IEC/IEEE 42010:2011



It is a set of **best practices** for the architectural description

WHAT MAKES UP AN AD?



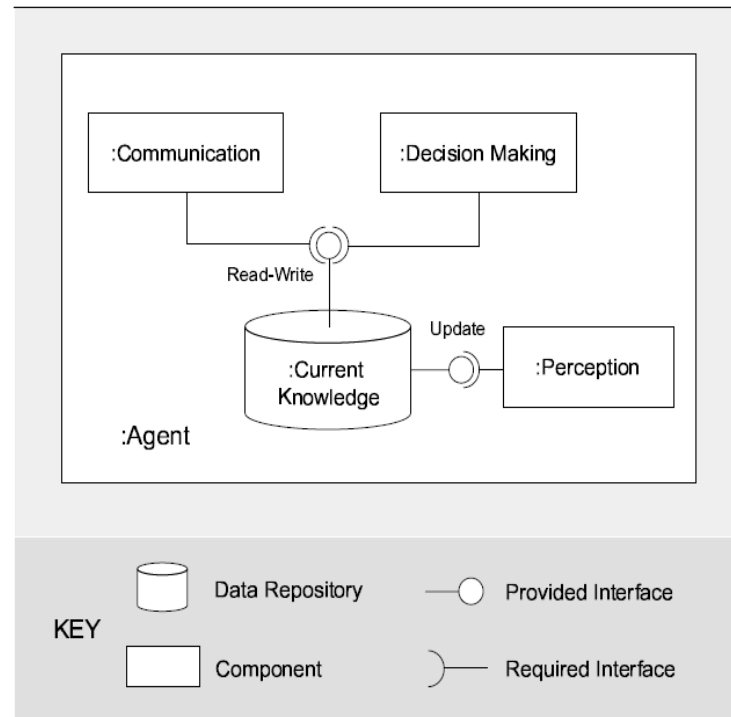
[ISO/IEC/IEEE 42010]

ARCHITECTURAL VIEW

- It is a **collection of models** representing the **architecture** of the whole system relative to **a set of concern**

[ISO/IEC/IEEE 42010]

- A structural view :



Shared data view of an agent
[Guessi; Damasceno, 2016]

VIEWPOINTS

- It is artifact that establishes conventions for creating, interpreting and use views
 - Example: language, notation, design rules, model types, modeling methods, analysis techniques, and any associated methods.
- It expresses **how** to construct **a view**
- It ensures that the concern of **stakeholders** are:
 - Identified, allocated and covered

[Guessi, 2013] [ISO/IEC/IEEE 42010]

VIEWPOINTS

- The 42010 has not a standard convention
 - ✓ flexible choice
 - ✓ Concern vary systems to systems
 - ✓ Better meet the concern of stakeholders
- One viewpoints should:
 - ✓ Define functions and methods associated
 - ✓ Many organizations already have these pre-established decisions
 - ✓ Based on the architect's experience
 - ✓ Example: structural, system, etc

Stakeholders and identified concern => viewpoint => view.

[ISO/IEC/IEEE 42010]

VIEW X VIEWPOINTS

- Metaphor:
 - view : viewpoint :: map : legend

A legend defines:

- the conventions used in preparing a map (such as its scale, colors and other symbology) to aid readers in interpreting that map as intended.

[ISO/IEC/IEEE 42010]

THE BENEFITS OF USING VIEWS AND VIEWPOINTS

- Separation of concerns
- Communication with stakeholders groups
- Management of complexity
- Improvement developer focus

[Rozanski and Woods, s.d.]

VIEWPOINTS PITFALLS

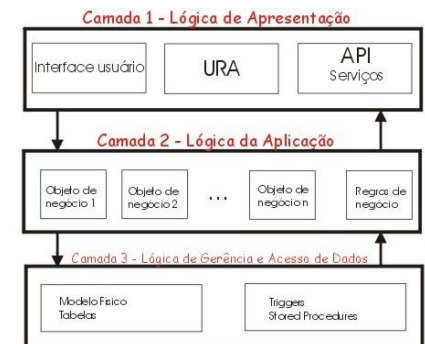
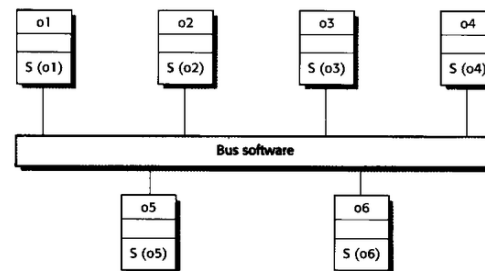
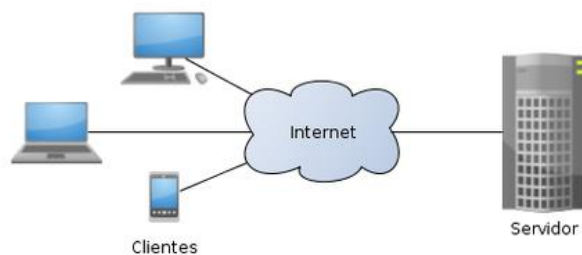
- Inconsistency
- Selection of the wrong set of views
- Fragmentation

[Rozanski and Woods, s.d.]

ARCHITECTURAL MODEL

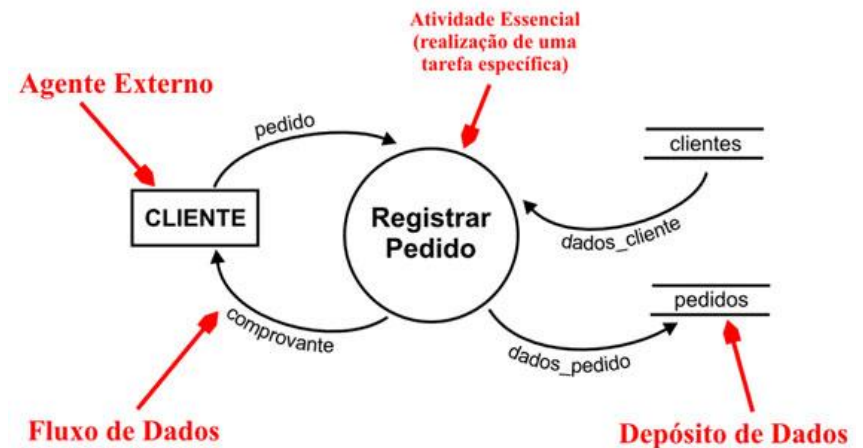
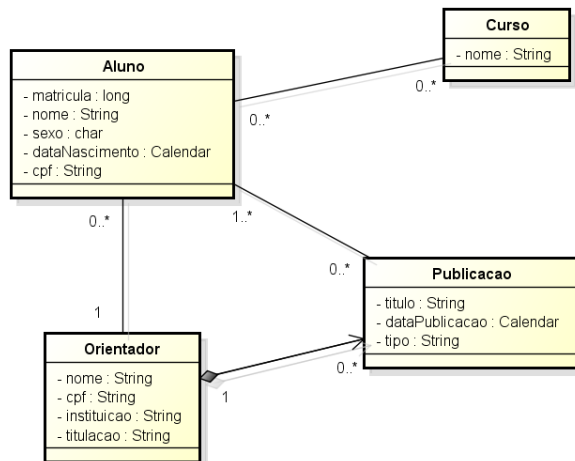
- A **view** is composed of **architecture models**
- Each model is constructed following **conventions established** by its **models kind**
- It can belong to **one or more views**
 - Example: **client-server, distributed objects, layers**

[ISO/IEC/IEEE 42010]



MODEL KIND

- It defines the conventions for one type of architecture model.
 - Example: meta-models



[ISO/IEC/IEEE 42010]

CORRESPONDENCE

- It expresses a relation between **AD Elements**.
- It are used to express architecture relations, such as:
 - composition, refinement, consistency, traceability, dependence, constraints and obligation

[ISO/IEC/IEEE 42010]

CORRESPONDENCE - EXAMPLE

Consider two views of a system S: a hardware view, HW(S), and a software component view, SC(S). Given that SC(S) includes software elements, e1, ... e6, and HW(S) includes hardware platforms, p1, ... p4, a correspondence expressing which software elements execute on which platforms.

(Element) ExecutesOn (Platforms) See rule: R1	
e1	p1, p4
e2	p2, p3
e3	p3
e4	p4

[ISO/IEC/IEEE 42010]

CORRESPONDENCE RULES

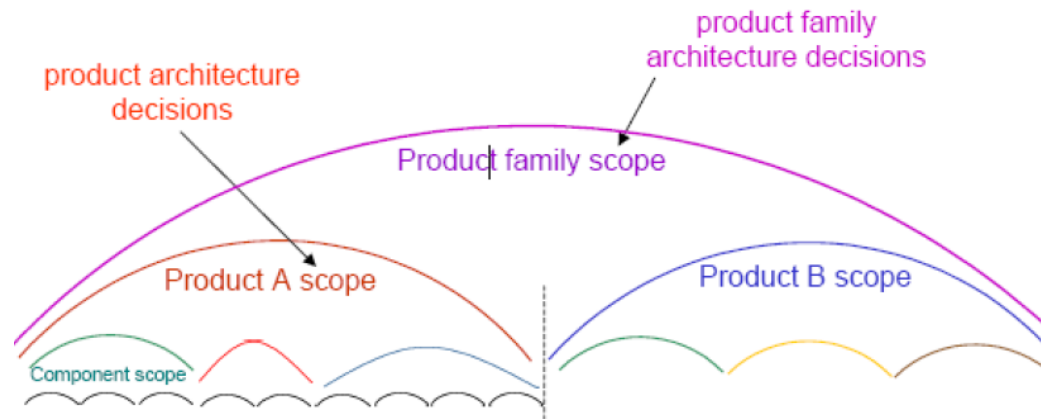
- It governs correspondence
- It expresses a constraint to be enforced on a correspondence
 - Example:

R1: Every software element, e_i , as defined by Software Components needs to execute on one or more platforms, p_j , as defined by Hardware.

[ISO/IEC/IEEE 42010]

ARCHITECTURE RATIONALE

- It records the explanation, justification or reasoning about particular decision
 - Example: reasons for the decision, alternatives and exchanges considered, possible consequences



Architecture is the set of decisions that cannot be delegated without compromising overall system objectives.

Copyright 1999-2005 Bredemeyer Consulting
<http://www.bredemeyer.com>

IndyJUG April 2005
WhatArchitectureDefinition.ppt Slide 11

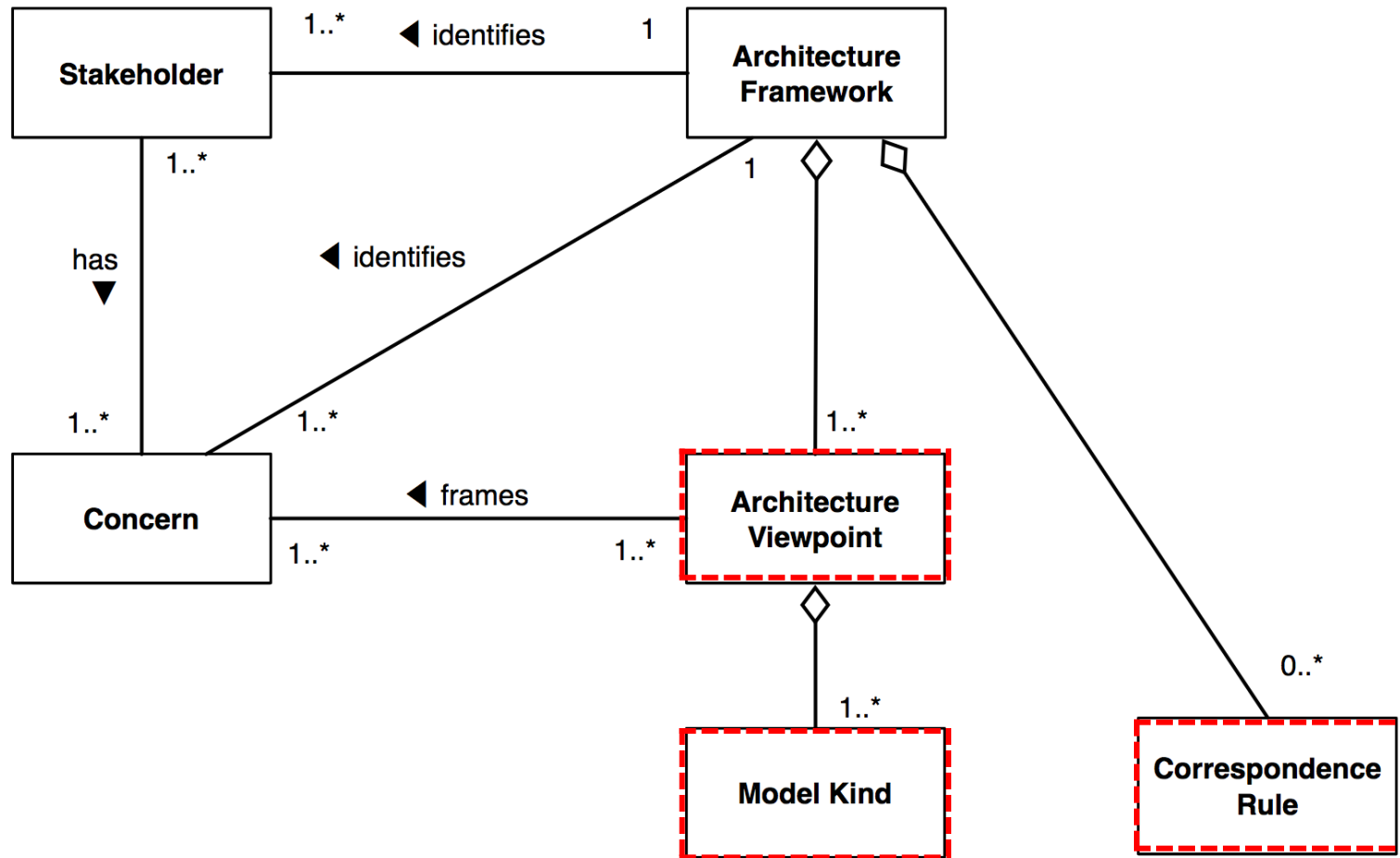
[ISO/IEC/IEEE 42010] [Guessi, 2013]

ARCHITECTURE FRAMEWORK

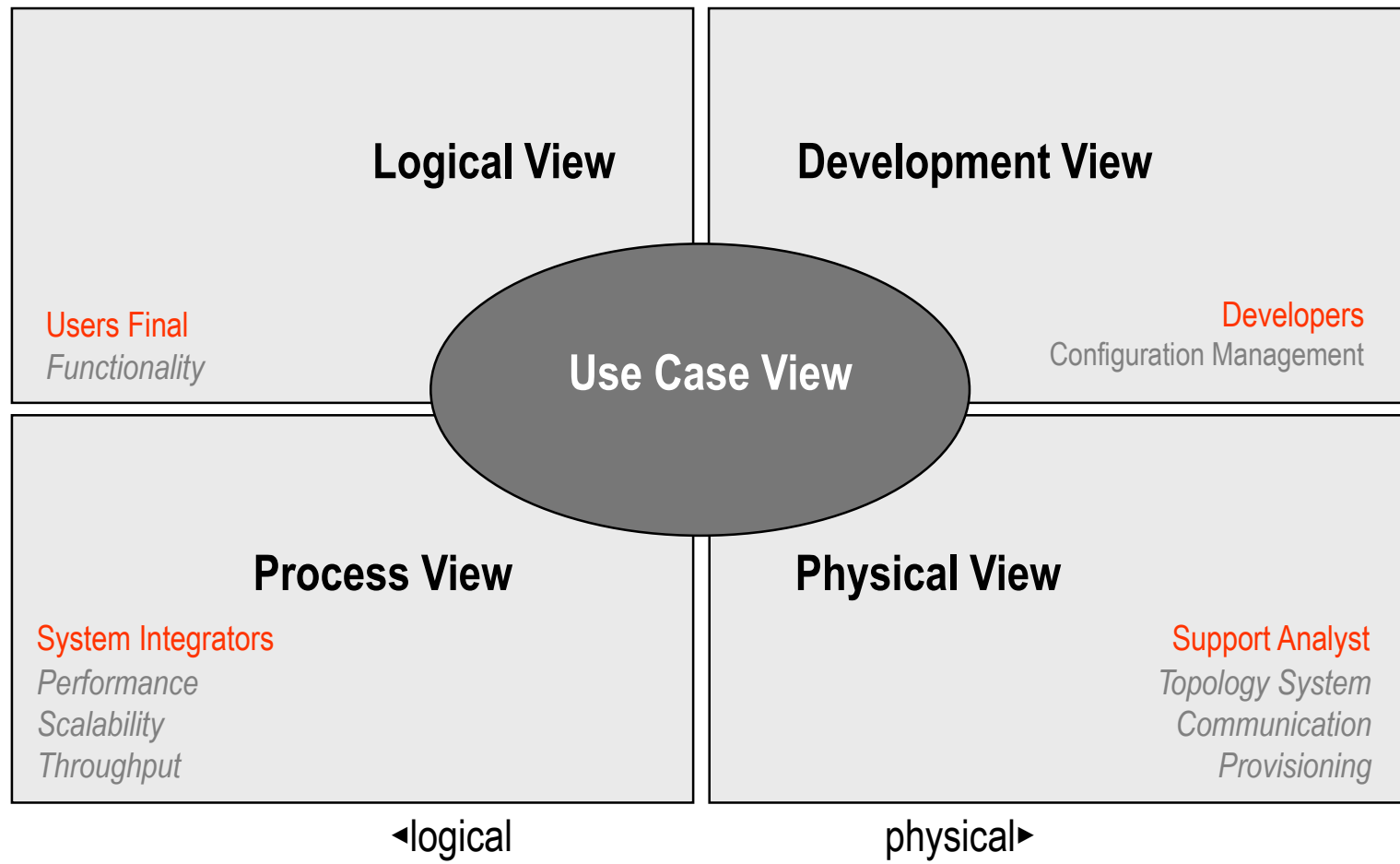
- It establishes a **common practice** for:
 - creating, interpreting, analyzing and using architecture descriptions
 - a particular domain of application or stakeholder community

- Federal Enterprise Architecture
- Kruchten's 4+1 Framework

ARCHITECTURE FRAMEWORK



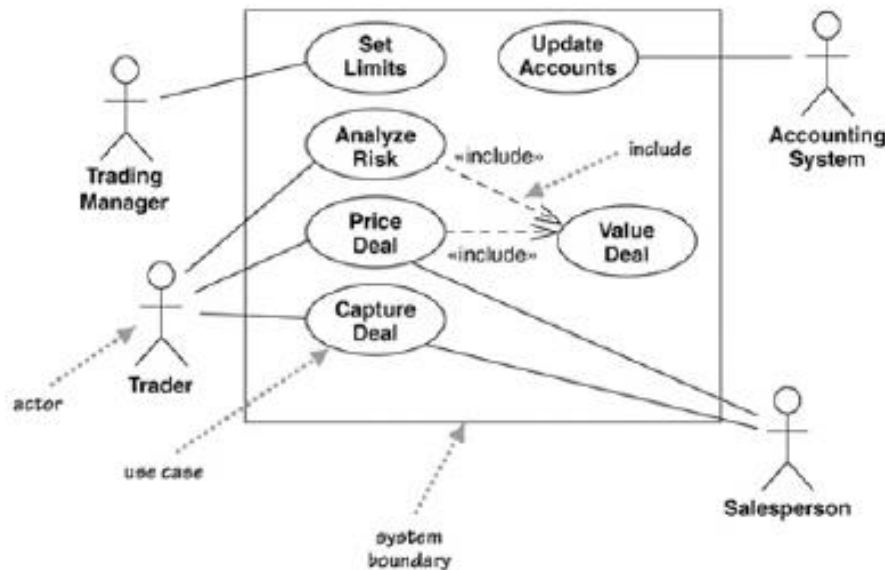
ARCHITECTURE FRAMEWORK– 4+1



FRAMEWORK ARQUITETURAL – 4+1

- Use Case View

- **Content:** software functionality, its external interfaces and key users
- **Diagrams:** use case diagrams
- **Interested:** users, systems of analysts, architects, developers

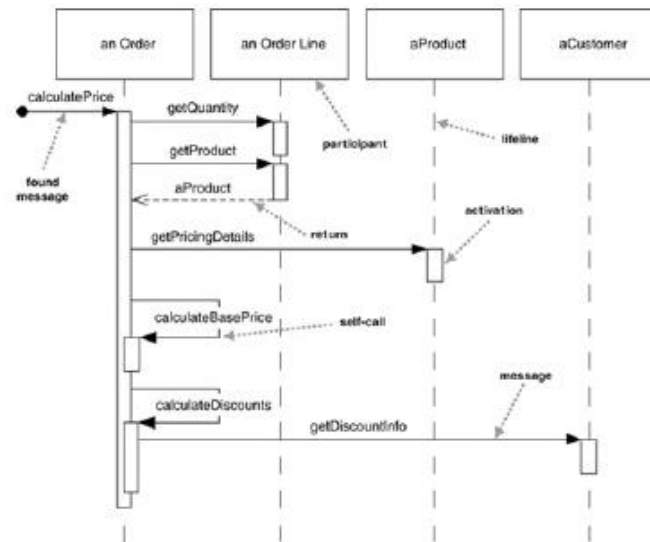
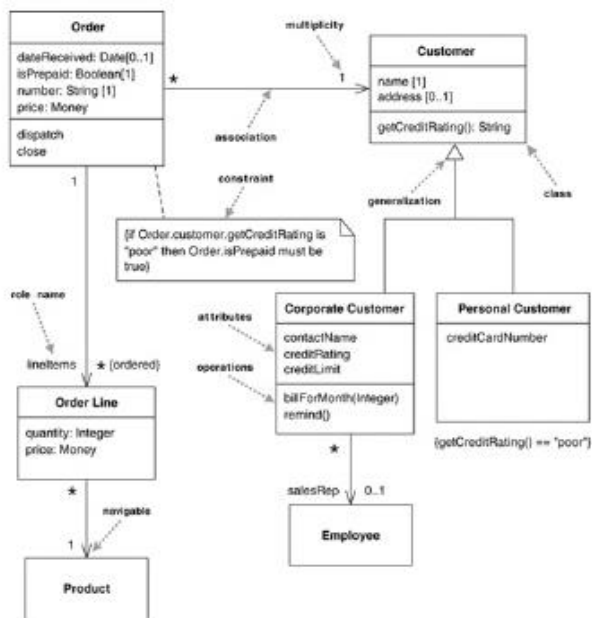


<http://www.institutogti.com.br>

FRAMEWORK ARQUITETURAL = 4+1

• Logical View

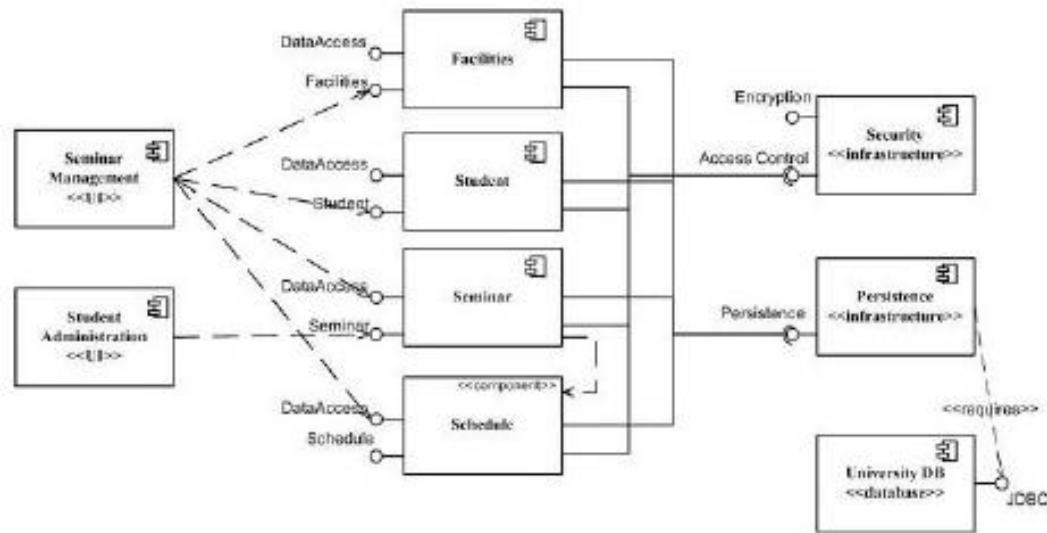
- **Content:** organizations of the elements of analysis and software design, and use case realizations
- **Diagrams:** class, sequence, communication, state
- **Interested:** users, systems of analysts, architects, developers



FRAMEWORK ARQUITETURAL – 4+1

• Development View

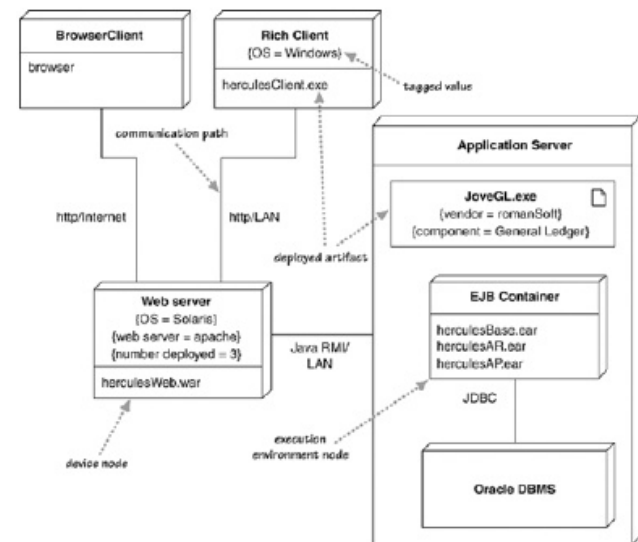
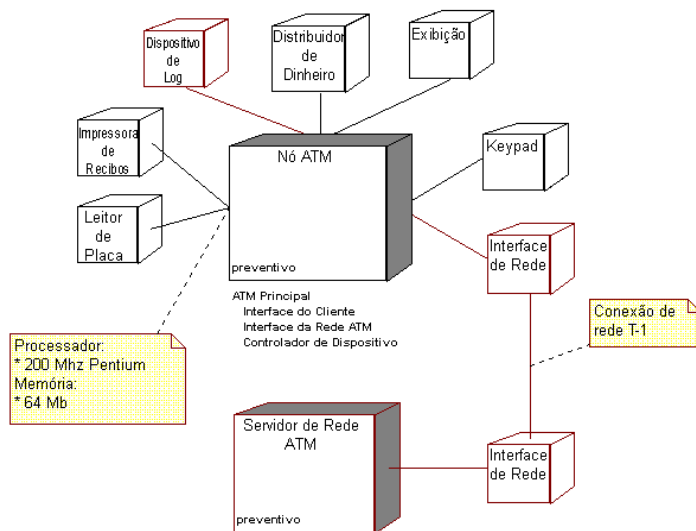
- **Content:** allocations of software design elements on components and subsystems and organizational components, subsystems and layers
- **Diagrams:** component and package diagrams
- **Interested:** architects, developers, integrators



FRAMEWORK ARQUITETURAL – 4+1

• Physical View

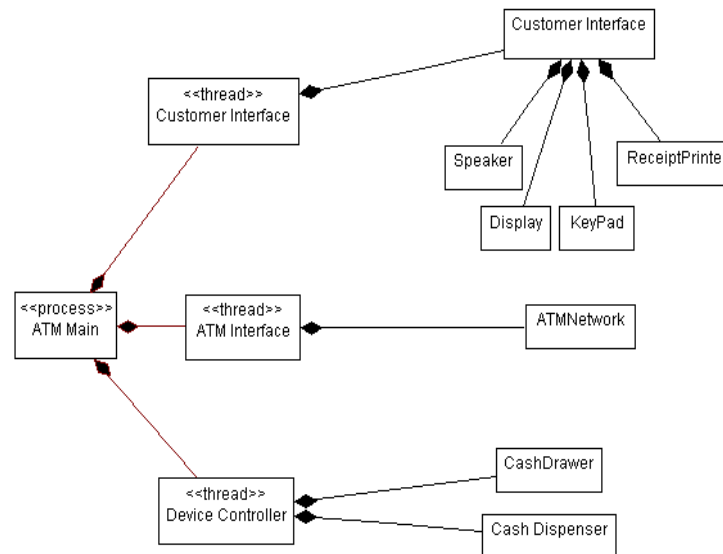
- **Content:** components allocation in artifacts, artifacts in the hardware and software infrastructure, and organization of hardware and software infrastructure
- **Diagrams:** deployment
- **Interested:** architects, integrators, support of analysts, operators



FRAMEWORK ARQUITETURAL – 4+1

• Process View

- **Content:** addresses the concurrent aspects and distribution, system integrity and fault-tolerance of a system at runtime
- **Diagrams:** Sequence, Communication, Activity, Activity, Timing and Interaction Overview
- **Interested:** architects, integrators, support analyst



ARCHITECTURE FRAMEWORK– 4+1

- What is use?
- Not all systems need all views:
 - Small systems (it discount the implementation view)
 - Single processor (it can discount the deployment)
 - Single process (it can discount the process view)
- Some systems require additional views:
 - Data view
 - Security view
 - Among others

FINAL CONSIDERATIONS

Architectural Description:

- are used by the parties that create, utilize and manage modern systems to improve communication and co-operation, enabling them to work in an integrated, coherent fashion
- is a work product which models the architecture of a system-of-interest
- have many uses by a variety of stakeholders throughout the system life cycle
- shall identify the system stakeholders having concerns considered fundamental to the architecture of the system-of-interest

INTERESTING LINKS

- [ISO-IEC-FDIS-42010](#)
- [Software Architecture Description & UML Workshop](#)
- [A Conceptual Model of Architecture Description](#)
- [Using the UML for Architectural Description](#)
- [System and Software Architecture Description Example](#)
- [Views and Viewpoints in Software Systems Architecture](#)
- [Documenting Software Architecting: View and Beyond](#)
- [Document Software Architecture](#)
- [Architectural Description Guideline](#)

BIBLIOGRAPHY

- Guessi, M. Subsídios para a representação de arquiteturas de referência de sistemas embarcados. Dissertation (Master) Institute of Mathematics and Computation Science – ICMC – University of São Paulo (USP), São Carlos, Brazil. 2013.
- Bass, L., Clements, P., & Kazman, R. (2003). Software Architecture in Practice. Addison-Wesley Professional, 2nd edition.
- ISO/IEC/IEEE 42010:2010 International Standard for Systems and Software Engineering -- Architectural description
- Rozanski, N. and Woods, E. Applying viewpoints and views to software engineering. Available in {http://www.viewpoints-and-perspectives.info/vpandp/wp-content/themes/secondedition/doc/VPandV_WhitePaper.pdf}. Access in 2016-03-29.

ARCHITECTURAL DESCRIPTION



Cristiane Lana e Délcio Silva

Software Architecture - SSC9441

Prof. Dr. Elisa Yumi Nakagawa