

02291 System Integration

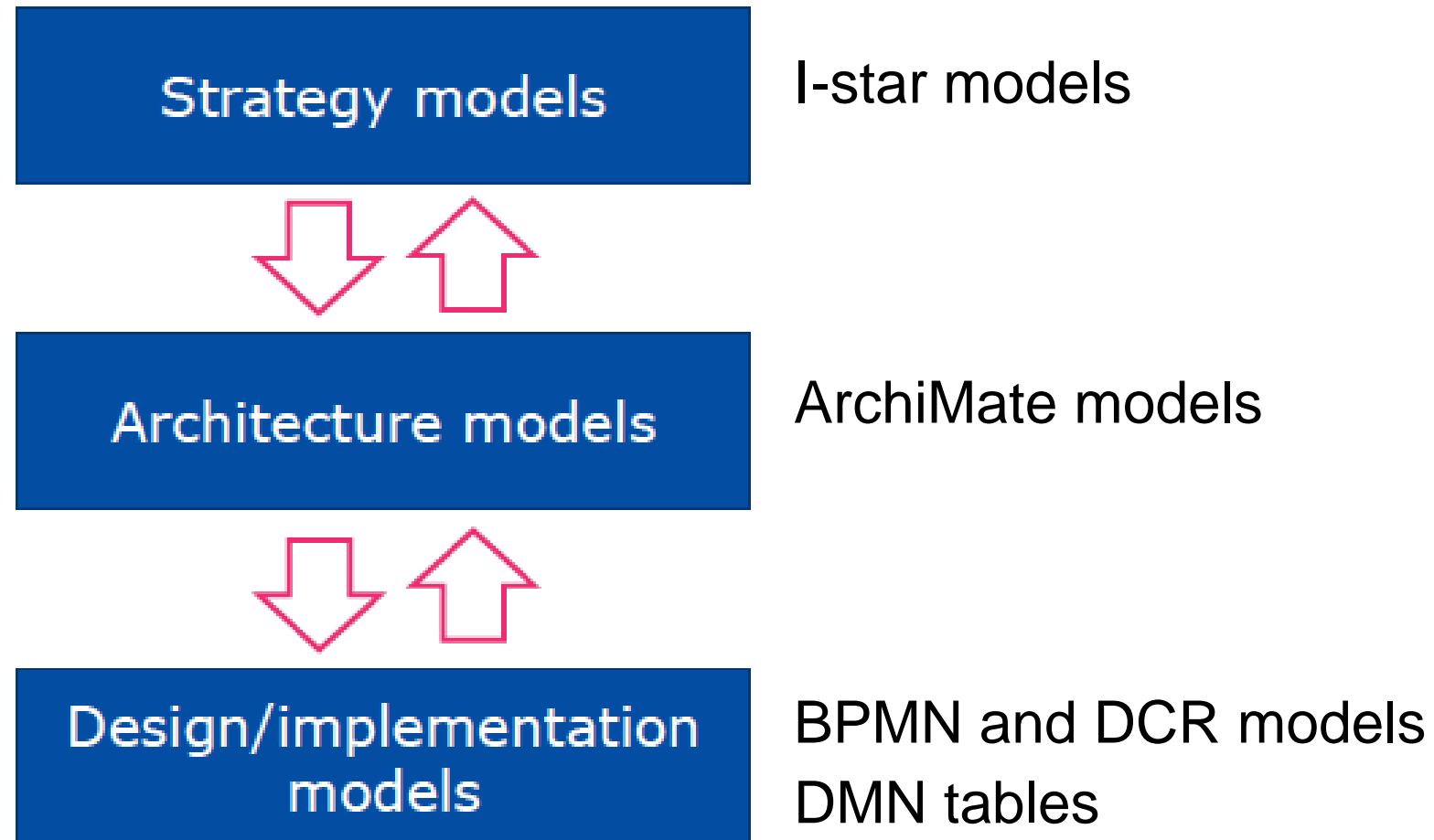
# Introduction to ArchiMate

© Giovanni Meroni

# What it is

- Archimate is a language for enterprise architecture modeling
- Mainly used to describe building blocks
- It is extensible

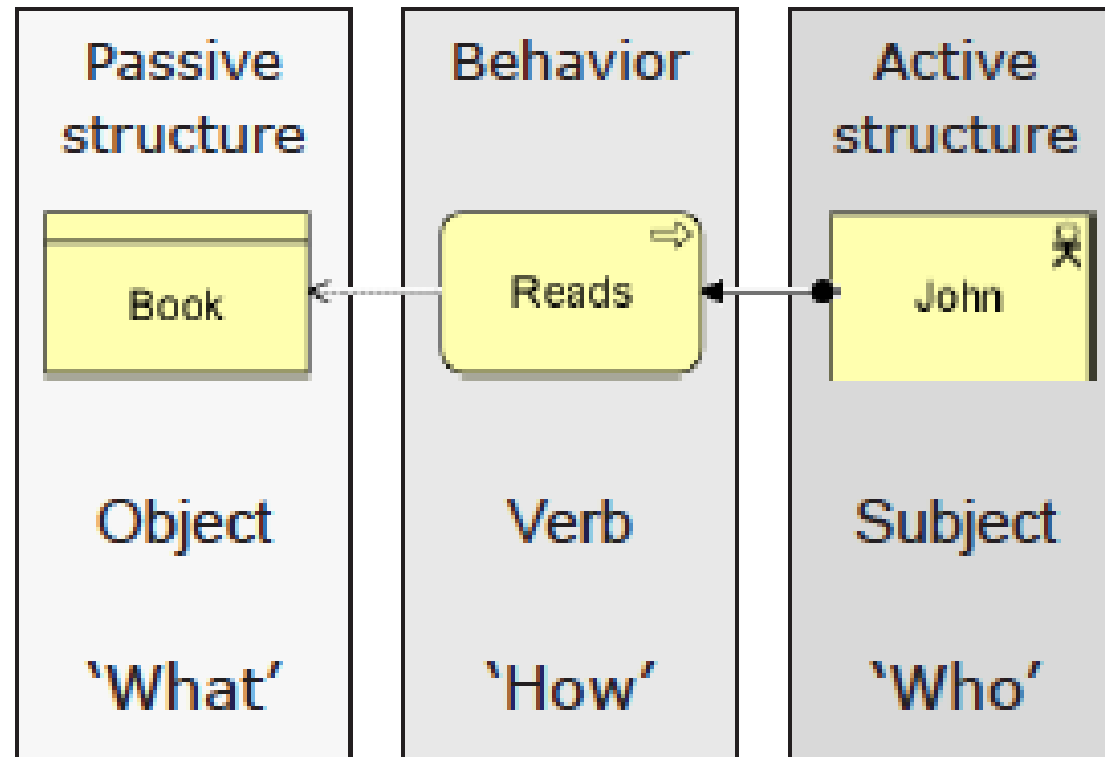
# Positioning ArchiMate



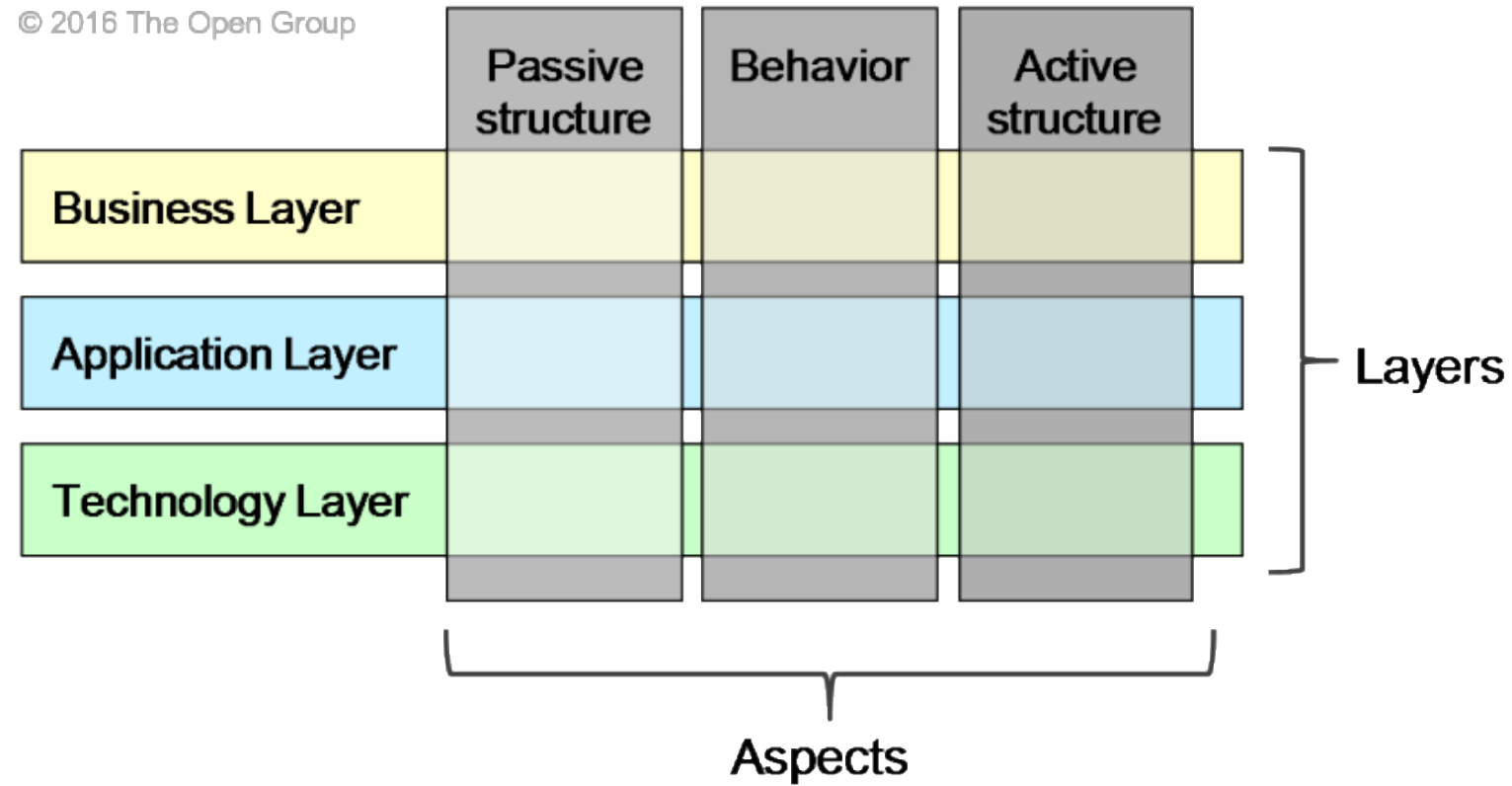
# Layers

- Three main layers of modeling
  - Business layer: offer products and services to external costumers
  - Application layer: support the business layer with application services – realized by software applications
  - Technology layer: offers infrastructure services needed to run applications, realized by computer and communication

# Aspects

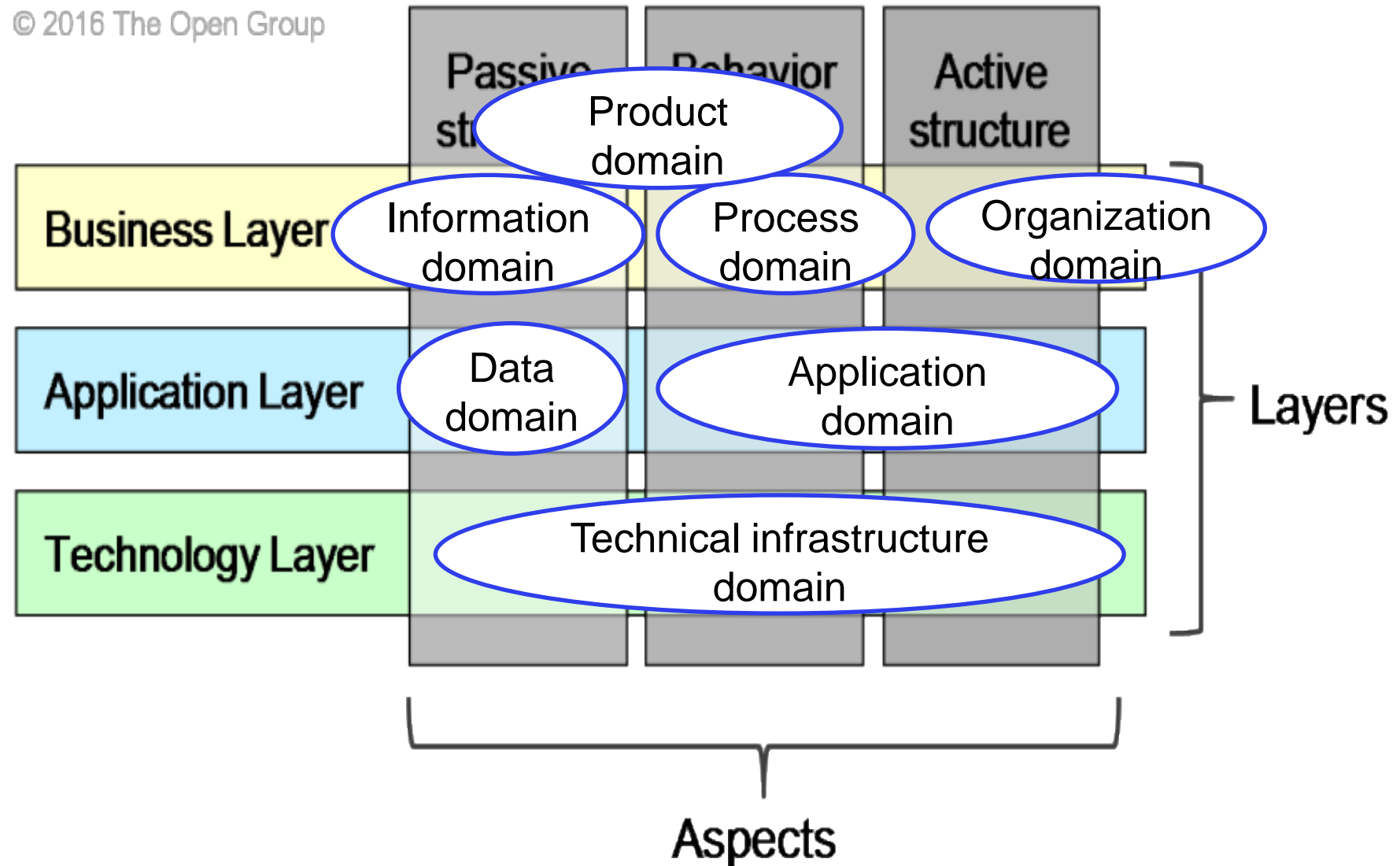








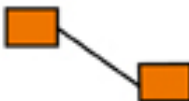

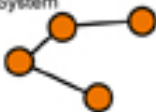





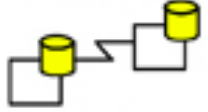


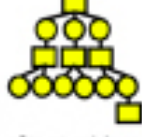
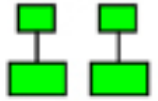
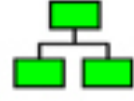










# Aspects-layers matrix



# Aspects-layers matrix




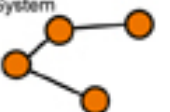


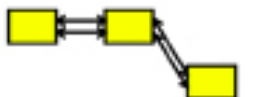

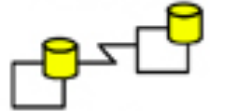




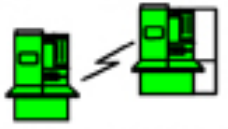
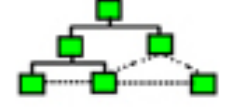






© 2016 The Open Group



	DATA <i>What</i>	FUNCTION <i>How</i>	NETWORK <i>Where</i>	PEOPLE <i>Who</i>	TIME <i>When</i>	MOTIVATION <i>Why</i>	
SCOPE (CONTEXTUAL)	List of Things Important to the Business 	List of Processes the Business Performs 	List of Locations in which the Business Operates 	List of Organizations Important to the Business 	List of Events/Cycles Significant to the Business 	List of Business Goals/Strategies 	SCOPE (CONTEXTUAL)
<i>Planner</i>	ENTITY = Class of Business Thing	Process = Class of Business Process	Node = Major Business Location	People = Major Organization Unit	Time = Major Business Event/Cycle	Ends/Mean = Major Business Goal/Strategy	<i>Planner</i>
BUSINESS MODEL (CONCEPTUAL)	e.g. Semantic Model 	e.g. Business Process Model 	e.g. Business Logistics System 	e.g. Work Flow Model 	e.g. Master Schedule 	e.g. Business Plan 	BUSINESS MODEL (CONCEPTUAL)
<i>Owner</i>	Ent = Business Entity Rein = Business Relationship	Proc. = Business Process I/O = Business Resources	Node = Business Location Link = Business Linkage	People = Organization Unit Work = Work Product	Time = Business Event Cycle = Business Cycle	End = Business Objective Means = Business Strategy	<i>Owner</i>
SYSTEM MODEL (LOGICAL)	e.g. Logical Data Model 	e.g. Application Architecture 	e.g. Distributed System Architecture 	e.g. Human Interface Architecture 	e.g. Processing Structure 	e.g. Business Rule Model 	SYSTEM MODEL (LOGICAL)
<i>Designer</i>	Ent = Data Entity Rein = Data Relationship	Proc. = Application Function I/O = User Views	Node = I/S Function (Processor, Storage, etc) Link = Line Characteristics	People = Role Work = Deliverable	Time = System Event Cycle = Processing Cycle	End = Structural Assertion Means = Action Assertion	<i>Designer</i>
TECHNOLOGY MODEL (PHYSICAL)	e.g. Physical Data Model 	e.g. System Design 	e.g. Technology Architecture 	e.g. Presentation Architecture 	e.g. Control Structure 	e.g. Rule Design 	TECHNOLOGY MODEL (PHYSICAL)
<i>Builder</i>	Ent = Segment/Table/etc. Rein = Pointer/Key/etc.	Proc. = Computer Function I/O = Data Elements/Sets	Node = Hardware/Systems Software Link = Line Specifications	People = User Work = Screen Format	Time = Execute Cycle = Component Cycle	End = Condition Means = Action	<i>Builder</i>
DETAILED REPRESENTATIONS (OUT-OF-CONTEXT)	e.g. Data Definition 	e.g. Program 	e.g. Network Architecture 	e.g. Security Architecture 	e.g. Timing Definition 	e.g. Rule Specification 	DETAILED REPRESENTATIONS (OUT-OF-CONTEXT)
<i>Sub-Contractor</i>	Ent = Field Rein = Address	Proc. = Language Statement I/O = Control Block	Node = Address Link = Protocol	People = Identity Work = Job	Time = Interrupt Cycle = Machine Cycle	End = Sub-condition Means = Stop	<i>Sub-Contractor</i>
FUNCTIONING ENTERPRISE	e.g. DATA	e.g. FUNCTION	e.g. NETWORK	e.g. ORGANIZATION	e.g. SCHEDULE	e.g. STRATEGY	FUNCTIONING ENTERPRISE



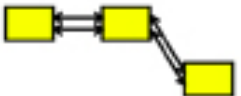

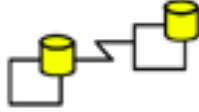


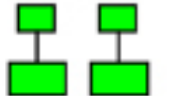









© John A. Zachman, Zachman International






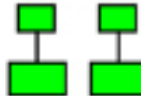



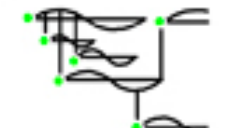





	DATA <i>What</i>	FUNCTION <i>How</i>	NETWORK <i>Where</i>	PEOPLE <i>Who</i>	TIME <i>When</i>	MOTIVATION <i>Why</i>	
SCOPE (CONTEXTUAL)	Business					List of Events/Cycles Significant to the Business  Time = Major Business Event/Cycle	SCOPE (CONTEXTUAL)
Planner							Planner
BUSINESS MODEL (CONCEPTUAL)	e.g. Semantic Model  Ent = Business Entity Rein = Business Relationship	e.g. Business Process Model  Proc. = Business Process I/O = Business Resources	e.g. Business Logistics System  Node = Business Location Link = Business Linkage	e.g. Work Flow Model  People = Organization Unit Work = Work Product	e.g. Master Schedule  Time = Business Event Cycle = Business Cycle	BUSINESS MODEL (CONCEPTUAL)	Owner
SYSTEM MODEL (LOGICAL)	e.g. Logical Data Model  Ent = Data Entity Rein = Data Relationship	e.g. Application Architecture  Proc. = Application Function I/O = User Views	e.g. Distributed System Architecture  Node = I/S Function (Processor, Storage, etc) Link = Line Characteristics	e.g. Human Interface Architecture  People = Role Work = Deliverable	e.g. Processing Structure  Time = System Event Cycle = Processing Cycle	SYSTEM MODEL (LOGICAL)	Designer
TECHNOLOGY MODEL (PHYSICAL)	e.g. Physical Data Model  Ent = Segment/Table/etc. Rein = Pointer/Key/etc.	e.g. System Design  Proc. = Computer Function I/O = Data Elements/Sets	e.g. Technology Architecture  Node = Hardware/Systems Software Link = Line Specifications	e.g. Presentation Architecture  People = User Work = Screen Format	e.g. Control Structure  Time = Execute Cycle = Component Cycle	TECHNOLOGY MODEL (PHYSICAL)	Builder
DETAILED REPRESENTATIONS (OUT-OF-CONTEXT)	e.g. Data Definition  Ent = Field Rein = Address	e.g. Program  Proc. = Language Statement I/O = Control Block	e.g. Network Architecture  Node = Address Link = Protocol	e.g. Security Architecture  People = Identity Work = Job	e.g. Timing Definition  Time = Interrupt Cycle = Machine Cycle	DETAILED REPRESENTATIONS (OUT-OF-CONTEXT)	Sub-Contractor
FUNCTIONING ENTERPRISE	e.g. DATA	e.g. FUNCTION	e.g. NETWORK	e.g. ORGANIZATION	e.g. SCHEDULE	e.g. STRATEGY	FUNCTIONING ENTERPRISE

Motivation




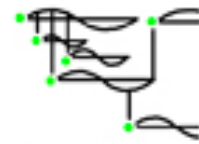





© John A. Zachman, Zachman International

	DATA <i>What</i>	FUNCTION <i>How</i>	NETWORK <i>Where</i>	PEOPLE <i>Who</i>	TIME <i>When</i>	MOTIVATION <i>Why</i>	
SCOPE (CONTEXTUAL)	Business					List of Events/Cycles Significant to the Business 	SCOPE (CONTEXTUAL)
<i>Planner</i>						Time = Major Business Event/Cycle	<i>Planner</i>
BUSINESS MODEL (CONCEPTUAL)						e.g. Master Schedule 	BUSINESS MODEL (CONCEPTUAL)
<i>Owner</i>						Time = Business Event Cycle = Business Cycle	<i>Owner</i>
SYSTEM MODEL (LOGICAL)	e.g. Logical Data Model  Ent = Data Entity Rein = Data Relationship	e.g. Application Architecture  Proc. = Application Function I/O = User Views	e.g. Distributed System Architecture  Node = I/S Function (Processor, Storage, etc) Link = Line Characteristics	e.g. Human Interface Architecture  People = Role Work = Deliverable	e.g. Processing Structure  Time = System Event Cycle = Processing Cycle	Motivation	SYSTEM MODEL (LOGICAL)
<i>Designer</i>							<i>Designer</i>
TECHNOLOGY MODEL (PHYSICAL)	e.g. Physical Data Model  Ent = Segment/Table/etc. Rein = Pointer/Key/etc.	e.g. System Design  Proc. = Computer Function I/O = Data Elements/Sets	e.g. Technology Architecture  Node = Hardware/Systems Software Link = Line Specifications	e.g. Presentation Architecture  People = User Work = Screen Format	e.g. Control Structure  Time = Execute Cycle = Component Cycle		TECHNOLOGY MODEL (PHYSICAL)
<i>Builder</i>							<i>Builder</i>
DETAILED REPRESENTATIONS (OUT-OF-CONTEXT)	e.g. Data Definition  Ent = Field Rein = Address	e.g. Program  Proc. = Language Statement I/O = Control Block	e.g. Network Architecture  Node = Address Link = Protocol	e.g. Security Architecture  People = Identity Work = Job	e.g. Timing Definition  Time = Interrupt Cycle = Machine Cycle		DETAILED REPRESENTATIONS (OUT-OF-CONTEXT)
<i>Sub-Contractor</i>							<i>Sub-Contractor</i>
FUNCTIONING ENTERPRISE	e.g. DATA	e.g. FUNCTION	e.g. NETWORK	e.g. ORGANIZATION	e.g. SCHEDULE	e.g. STRATEGY	FUNCTIONING ENTERPRISE









© John A. Zachman, Zachman International

	DATA <i>What</i>	FUNCTION <i>How</i>	NETWORK <i>Where</i>	PEOPLE <i>Who</i>	TIME <i>When</i>	MOTIVATION <i>Why</i>	
SCOPE (CONTEXTUAL)  <i>Planner</i>	Business				List of Events/Cycles Significant to the Business 	Motivation	SCOPE (CONTEXTUAL)  <i>Planner</i>
BUSINESS MODEL (CONCEPTUAL)  <i>Owner</i>					e.g. Master Schedule 		BUSINESS MODEL (CONCEPTUAL)  <i>Owner</i>
SYSTEM MODEL (LOGICAL)  <i>Designer</i>	Application				e.g. Processing Structure 		SYSTEM MODEL (LOGICAL)  <i>Designer</i>
TECHNOLOGY MODEL (PHYSICAL)  <i>Builder</i>	e.g. Physical Data Model  Ent = Segment/Table/etc. Rein = Pointer/Key/etc.	e.g. System Design  Proc. = Computer Function I/O = Data Elements/Sets	e.g. Technology Architecture  Node = Hardware/Systems Link = Line Specifications	e.g. Presentation Architecture  People = User Work = Screen Format	e.g. Control Structure  Time = Execute Cycle = Component Cycle		TECHNOLOGY MODEL (PHYSICAL)  <i>Builder</i>
DETAILED REPRESENTATIONS (OUT-OF-CONTEXT)  <i>Sub-Contractor</i>	e.g. Data Definition  Ent = Field Rein = Address	e.g. Program  Proc. = Language Statement I/O = Control Block	e.g. Network Architecture  Node = Address Link = Protocol	e.g. Security Architecture  People = Identity Work = Job	e.g. Timing Definition  Time = Interrupt Cycle = Machine Cycle		DETAILED REPRESENTATIONS (OUT-OF-CONTEXT)  <i>Sub-Contractor</i>
FUNCTIONING ENTERPRISE	e.g. DATA	e.g. FUNCTION	e.g. NETWORK	e.g. ORGANIZATION	e.g. SCHEDULE	e.g. STRATEGY	FUNCTIONING ENTERPRISE

© John A. Zachman, Zachman International










	DATA <i>What</i>	FUNCTION <i>How</i>	NETWORK <i>Where</i>	PEOPLE <i>Who</i>	TIME <i>When</i>	MOTIVATION <i>Why</i>	
SCOPE (CONTEXTUAL)  <i>Planner</i>	Business				List of Events/Cycles Significant to the Business 	Motivation	SCOPE (CONTEXTUAL)  <i>Planner</i>
BUSINESS MODEL (CONCEPTUAL)  <i>Owner</i>					Time = Major Business Event/Cycle e.g. Master Schedule 		BUSINESS MODEL (CONCEPTUAL)  <i>Owner</i>
SYSTEM MODEL (LOGICAL)  <i>Designer</i>	Application				e.g. Processing Structure 		SYSTEM MODEL (LOGICAL)  <i>Designer</i>
TECHNOLOGY MODEL (PHYSICAL)  <i>Builder</i>					e.g. Control Structure 		TECHNOLOGY MODEL (PHYSICAL)  <i>Builder</i>
DETAILED REPRESENTATIONS (OUT-OF-CONTEXT)  <i>Sub-Contractor</i>	e.g. Data Definition  Ent = Field Rein = Address	e.g. Program  Proc. = Language Statement I/O = Control Block	e.g. Network Architecture  Node = Address Link = Protocol	e.g. Security Architecture  People = Identity Work = Job	e.g. Timing Definition  Time = Interrupt Cycle = Machine Cycle		DETAILED REPRESENTATIONS (OUT-OF-CONTEXT)  <i>Sub-Contractor</i>
FUNCTIONING ENTERPRISE	e.g. DATA	e.g. FUNCTION	e.g. NETWORK	e.g. ORGANIZATION	e.g. SCHEDULE	e.g. STRATEGY	FUNCTIONING ENTERPRISE

© John A. Zachman, Zachman International




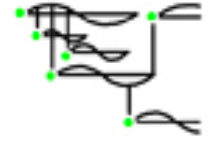





	DATA <i>What</i>	FUNCTION <i>How</i>	NETWORK <i>Where</i>	PEOPLE <i>Who</i>	TIME <i>When</i>	MOTIVATION <i>Why</i>	
SCOPE (CONTEXTUAL)  <i>Planner</i>	Passive	Business				List of Events/Cycles Significant to the Business 	SCOPE (CONTEXTUAL)  <i>Planner</i>
BUSINESS MODEL (CONCEPTUAL)  <i>Owner</i>						e.g. Master Schedule  Time = Business Event Cycle = Business Cycle	BUSINESS MODEL (CONCEPTUAL)  <i>Owner</i>
SYSTEM MODEL (LOGICAL)  <i>Designer</i>		Application				e.g. Processing Structure  Time = System Event Cycle = Processing Cycle	SYSTEM MODEL (LOGICAL)  <i>Designer</i>
TECHNOLOGY MODEL (PHYSICAL)  <i>Builder</i>						Technology	
DETAILED REPRESENTATIONS (OUT-OF-CONTEXT)  <i>Sub-Contractor</i>	e.g. Data Definition  Ent = Field Rein = Address	e.g. Program  Proc = Language Statement I/O = Control Block	e.g. Network Architecture  Node = Address Link = Protocol	e.g. Security Architecture  People = Identity Work = Job	e.g. Timing Definition  Time = Interrupt Cycle = Machine Cycle		
FUNCTIONING ENTERPRISE	e.g. DATA	e.g. FUNCTION	e.g. NETWORK	e.g. ORGANIZATION	e.g. SCHEDULE	e.g. STRATEGY	FUNCTIONING ENTERPRISE

© John A. Zachman, Zachman International



	DATA <i>What</i>	FUNCTION <i>How</i>	NETWORK <i>Where</i>	PEOPLE <i>Who</i>	TIME <i>When</i>	MOTIVATION <i>Why</i>	
SCOPE (CONTEXTUAL)  <i>Planner</i>	Passive	Behavior	Business			List of Events/Cycles Significant to the Business 	SCOPE (CONTEXTUAL)  <i>Planner</i>
BUSINESS MODEL (CONCEPTUAL)  <i>Owner</i>						e.g. Master Schedule  Time = Business Event Cycle = Business Cycle	BUSINESS MODEL (CONCEPTUAL)  <i>Owner</i>
SYSTEM MODEL (LOGICAL)  <i>Designer</i>	Application	Application	Application			e.g. Processing Structure  Time = System Event Cycle = Processing Cycle	SYSTEM MODEL (LOGICAL)  <i>Designer</i>
TECHNOLOGY MODEL (PHYSICAL)  <i>Builder</i>						e.g. Control Structure  Time = Execute Cycle = Component Cycle	TECHNOLOGY MODEL (PHYSICAL)  <i>Builder</i>
DETAILED REPRESENTATIONS (OUT-OF-CONTEXT)  <i>Sub-Contractor</i>	e.g. Data Definition  Ent = Field Rein = Address	e.g. Program  Proc = Language Statement I/O = Control Block	e.g. Network Architecture  Node = Address Link = Protocol	e.g. Security Architecture  People = Identity Work = Job	e.g. Timing Definition  Time = Interrupt Cycle = Machine Cycle	Motivation	DETAILED REPRESENTATIONS (OUT-OF-CONTEXT)  <i>Sub-Contractor</i>
FUNCTIONING ENTERPRISE	e.g. DATA	e.g. FUNCTION	e.g. NETWORK	e.g. ORGANIZATION	e.g. SCHEDULE		e.g. STRATEGY

© John A. Zachman, Zachman International

	DATA <i>What</i>	FUNCTION <i>How</i>	NETWORK <i>Where</i>	PEOPLE <i>Who</i>	TIME <i>When</i>	MOTIVATION <i>Why</i>	
SCOPE (CONTEXTUAL)  <i>Planner</i>	Passive	Behavior	Active		List of Events/Cycles Significant to the Business 	Motivation	SCOPE (CONTEXTUAL)  <i>Planner</i>
BUSINESS MODEL (CONCEPTUAL)  <i>Owner</i>					Time = Major Business Event/Cycle e.g. Master Schedule 		BUSINESS MODEL (CONCEPTUAL)  <i>Owner</i>
SYSTEM MODEL (LOGICAL)  <i>Designer</i>	Application	Application	Application		e.g. Processing Structure 		SYSTEM MODEL (LOGICAL)  <i>Designer</i>
TECHNOLOGY MODEL (PHYSICAL)  <i>Builder</i>					e.g. Control Structure 		TECHNOLOGY MODEL (PHYSICAL)  <i>Builder</i>
DETAILED REPRESENTATIONS (OUT-OF-CONTEXT)  <i>Sub-Contractor</i>	e.g. Data Definition  Ent = Field Rein = Address	e.g. Program  Proc. = Language Statement I/O = Control Block	e.g. Network Architecture  Node = Address Link = Protocol	e.g. Security Architecture  People = Identity Work = Job	e.g. Timing Definition  Time = Interrupt Cycle = Machine Cycle		DETAILED REPRESENTATIONS (OUT-OF-CONTEXT)  <i>Sub-Contractor</i>
FUNCTIONING ENTERPRISE	e.g. DATA	e.g. FUNCTION	e.g. NETWORK	e.g. ORGANIZATION	e.g. SCHEDULE	e.g. STRATEGY	FUNCTIONING ENTERPRISE

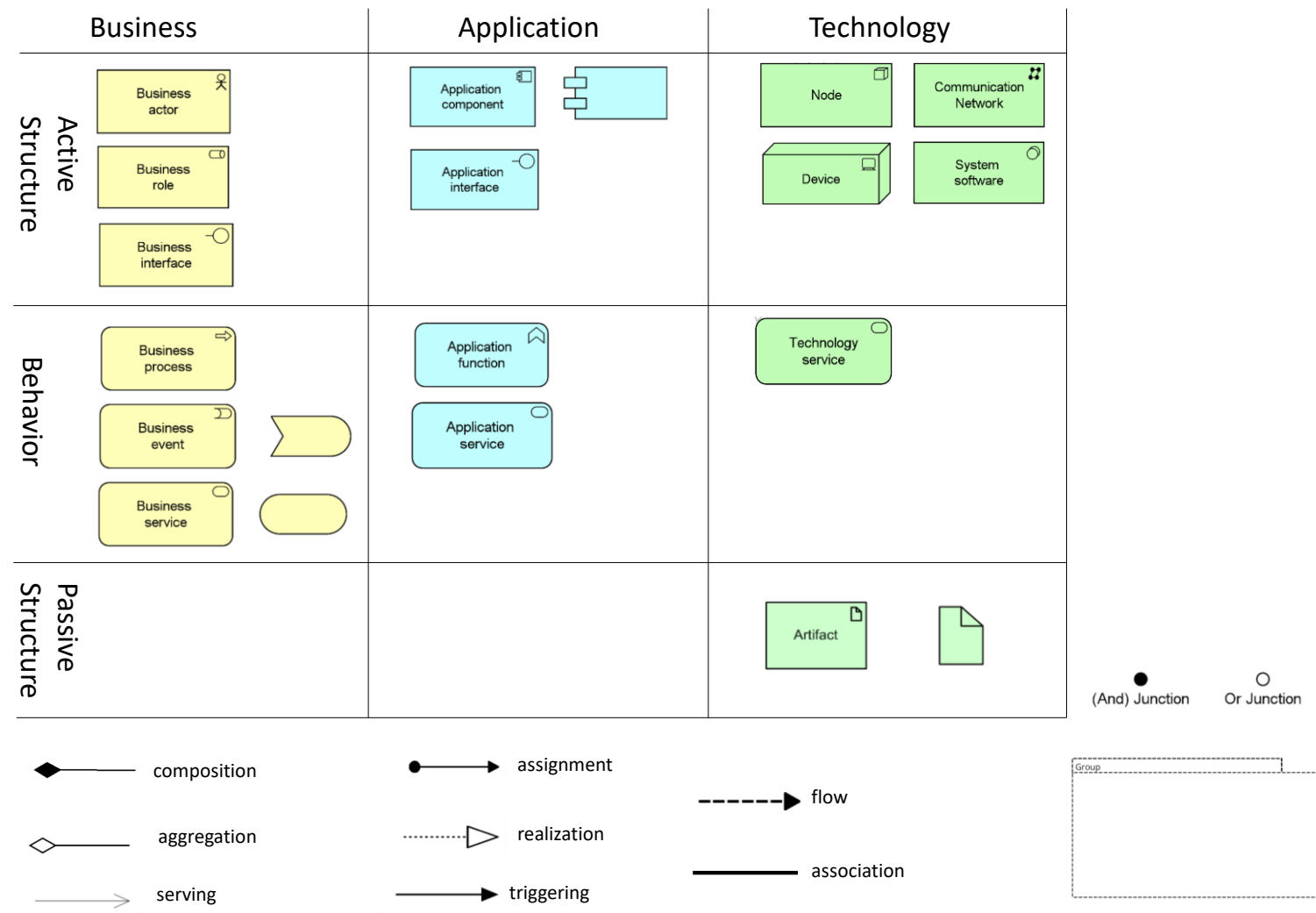
© John A. Zachman, Zachman International

	DATA <i>What</i>	FUNCTION <i>How</i>	NETWORK <i>Where</i>	PEOPLE <i>Who</i>	TIME <i>When</i>	MOTIVATION <i>Why</i>	
SCOPE (CONTEXTUAL)  <i>Planner</i>	Passive	Behavior	Active   <b>Business</b>			Motivation	SCOPE (CONTEXTUAL)  <i>Planner</i>
BUSINESS MODEL (CONCEPTUAL)  <i>Owner</i>							BUSINESS MODEL (CONCEPTUAL)  <i>Owner</i>
SYSTEM MODEL (LOGICAL)  <i>Designer</i>			<b>Application</b>				SYSTEM MODEL (LOGICAL)  <i>Designer</i>
TECHNOLOGY MODEL (PHYSICAL)  <i>Builder</i>			<b>Technology</b>				TECHNOLOGY MODEL (PHYSICAL)  <i>Builder</i>
DETAILED REPRESENTATIONS (OUT-OF-CONTEXT)  <i>Sub-Contractor</i>	<b>Implementation &amp; migration</b>						DETAILED REPRESENTATIONS (OUT-OF-CONTEXT)  <i>Sub-Contractor</i>
FUNCTIONING ENTERPRISE							FUNCTIONING ENTERPRISE
	e.g. DATA	e.g. FUNCTION	e.g. NETWORK	e.g. ORGANIZATION	e.g. SCHEDULE	e.g. STRATEGY	

© John A. Zachman, Zachman International



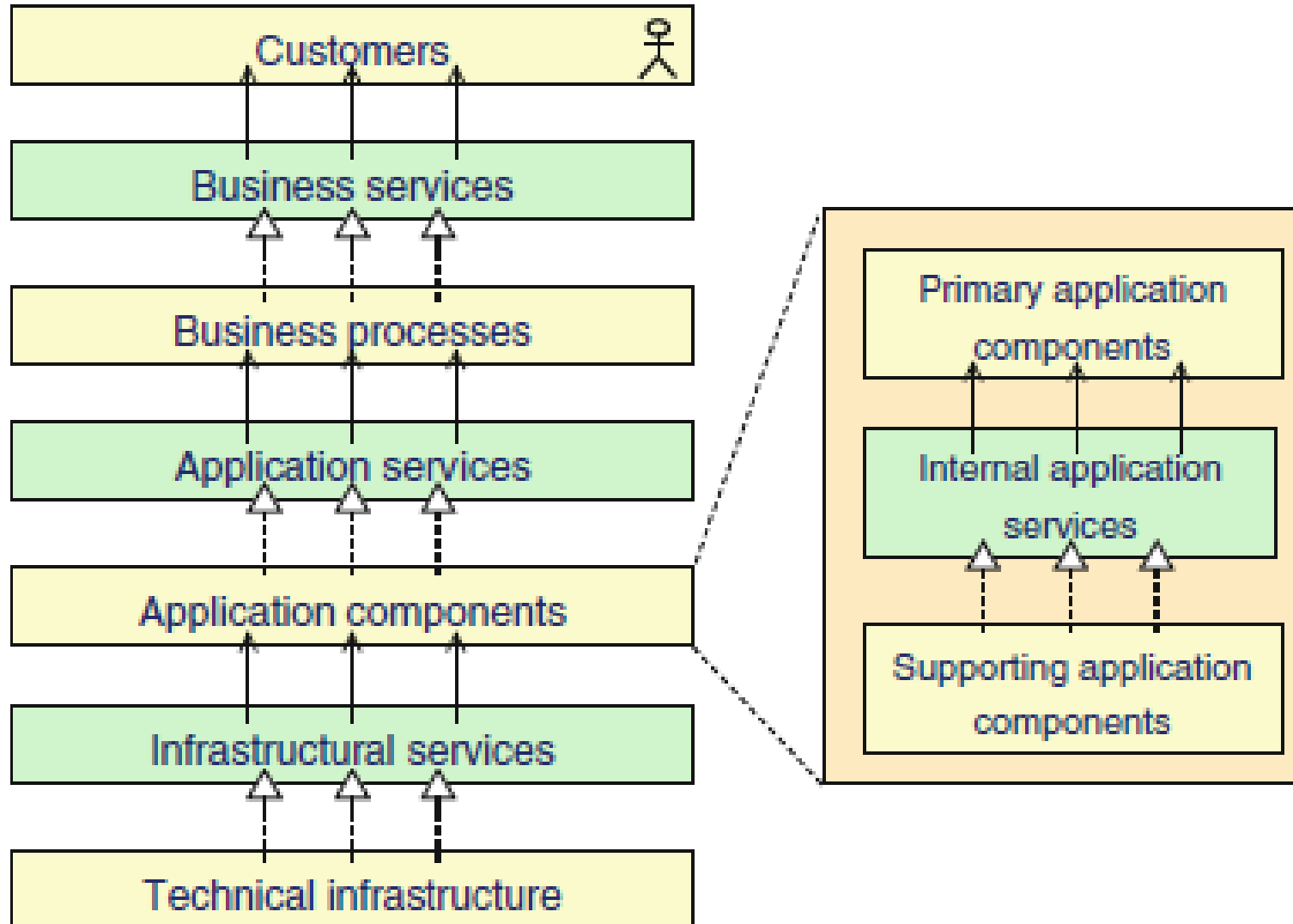
# ArchiMate cheat sheet



# Service Oriented approach

- Components (business, application, software, infrastructure) provide services to other components
- Services at all levels
- Examples:
  - Web services
  - In cloud computing: Software (SaaS), Platform (PaaS), Infrastructure (IaaS)
- Services are central to Archimate's architecture models

# Service Oriented approach



Lankhorst et al  
p. 78

# Business Layer

# Active elements

Business  
actor



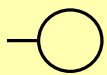
A business entity that is capable of performing behavior

Business  
role



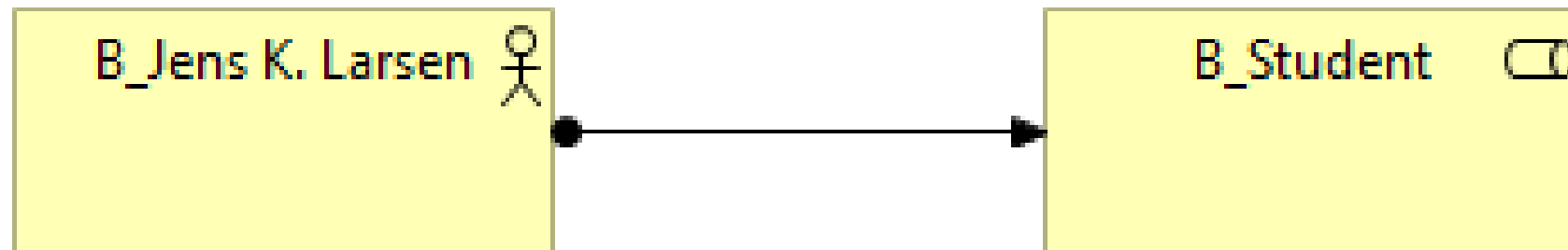
The responsibility for performing specific behavior, to which an action can be assigned

Business  
interface

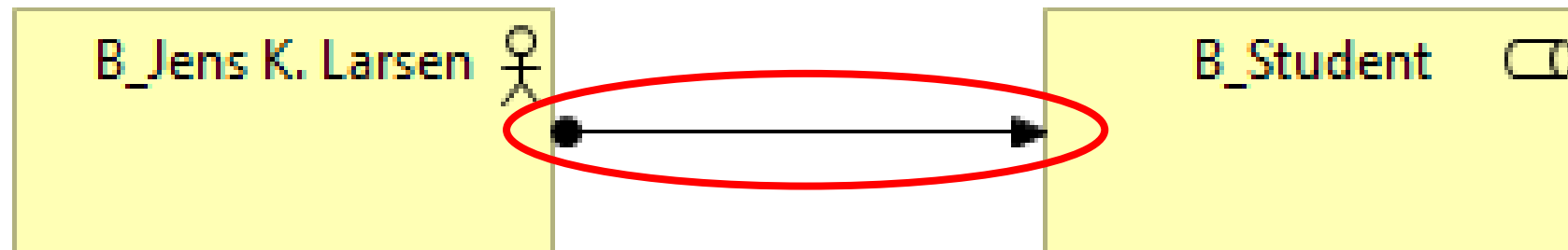


A point of access where a business service is made available to the environment

# Active elements: example

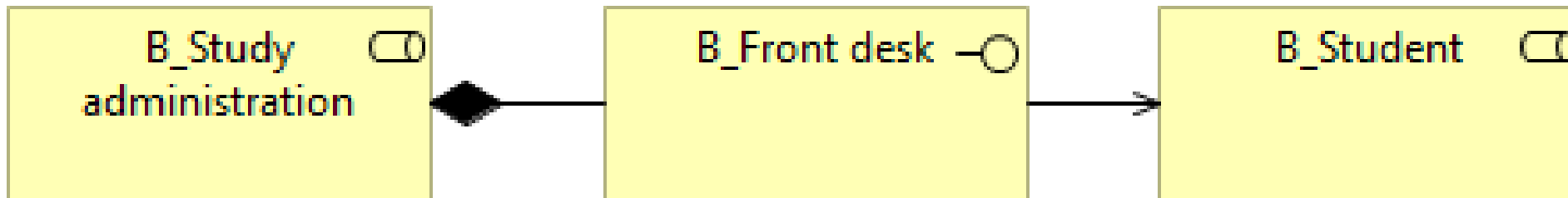


# Active elements: example



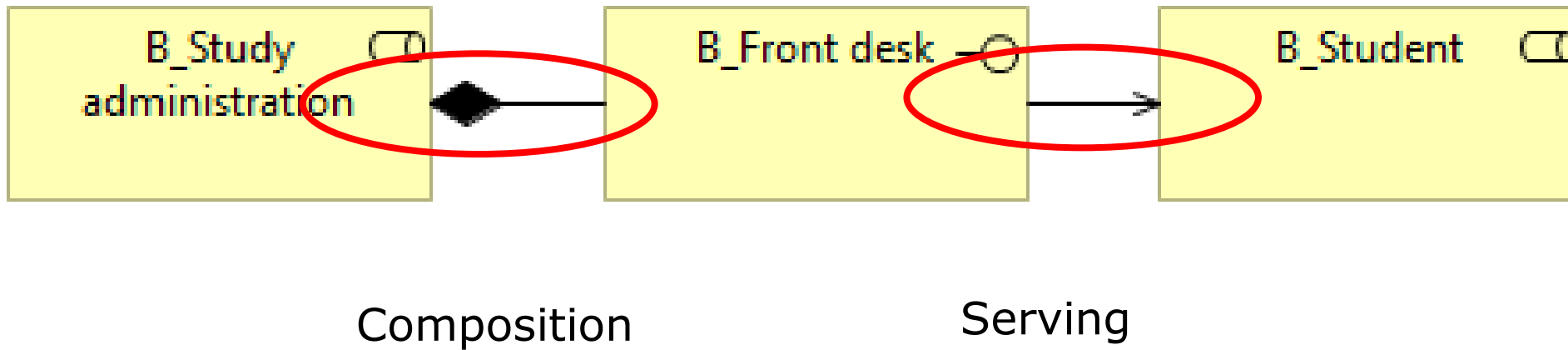
Assignment

# Active elements: example





# Active elements: example



# Behavior elements

Business  
process



A sequence of business behaviors that achieves a specific outcome

Business  
function



A collection of business behaviors based on a chosen set of criteria, aligned to an organization

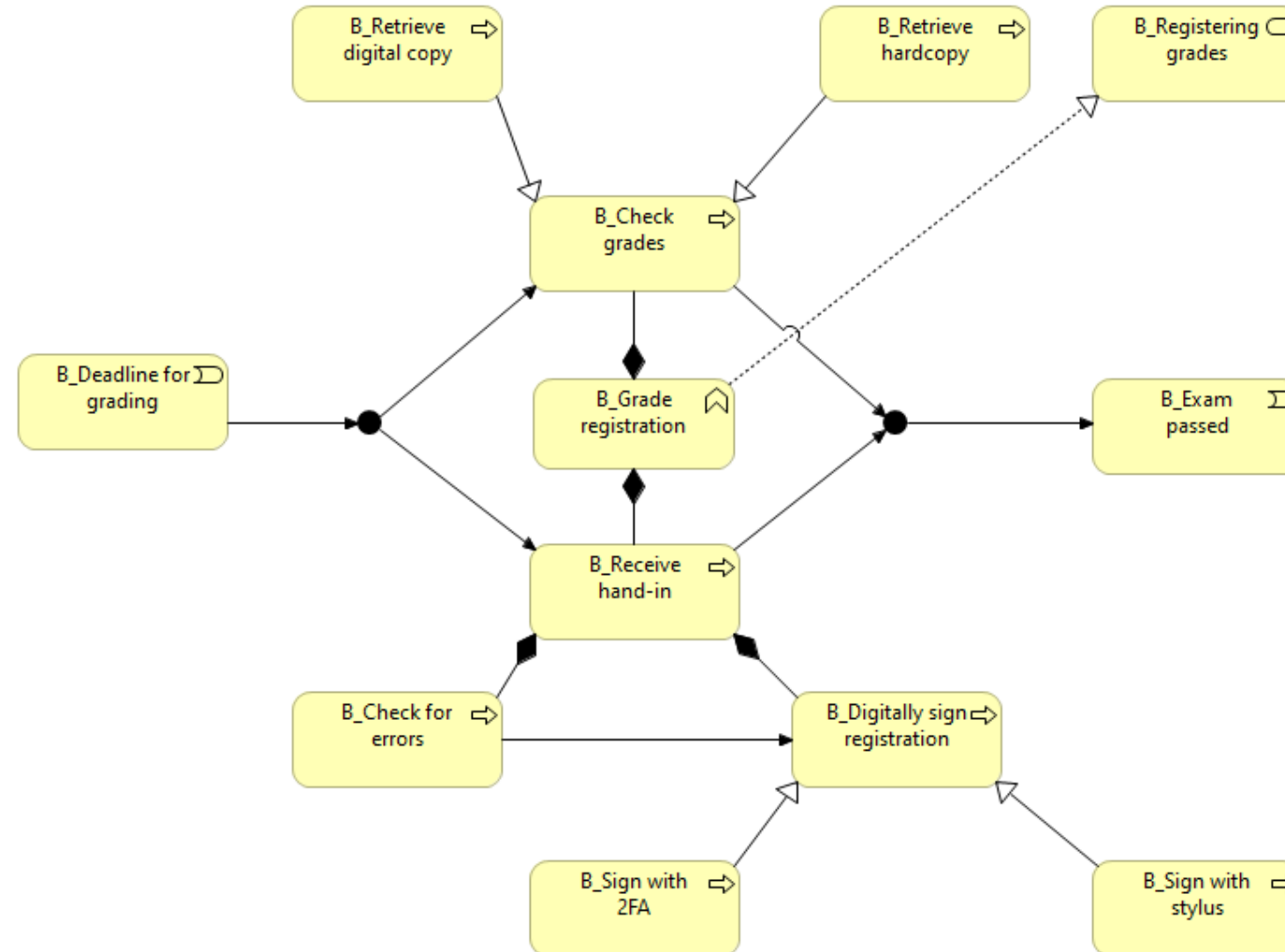
Business  
service

An explicitly defined exposed business behavior

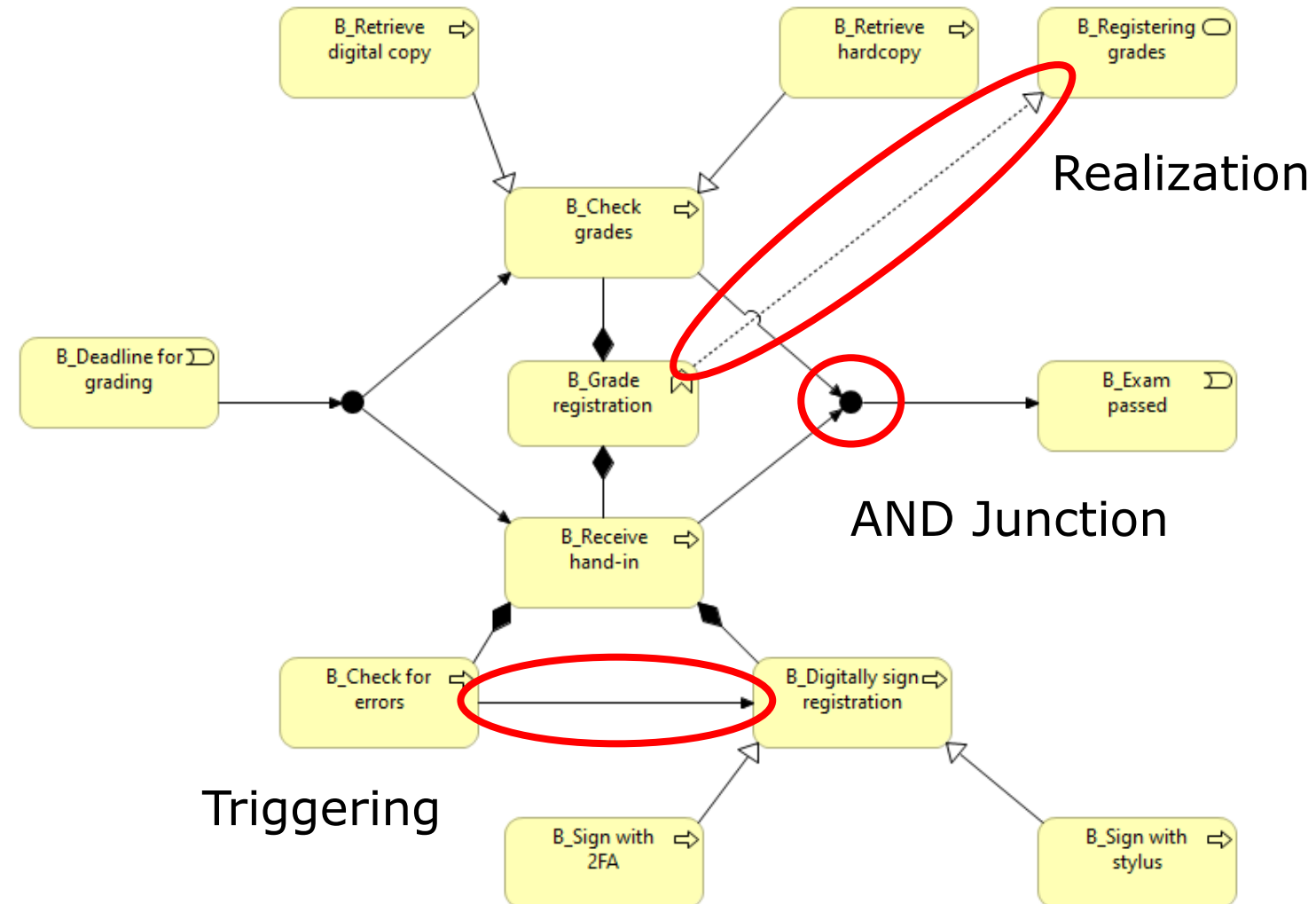
Business  
event

A business behavior element that denotes an organizational state change.

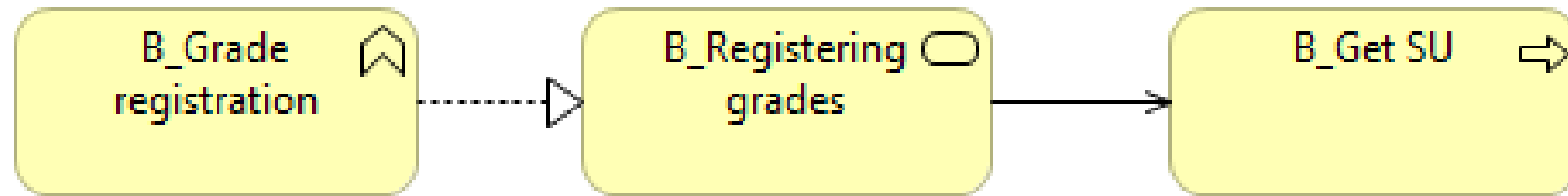
# Behavior elements: example



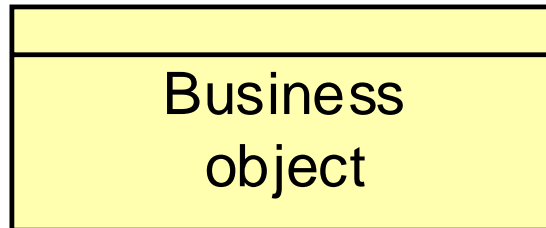
# Behavior elements: example



# Behavior elements: example

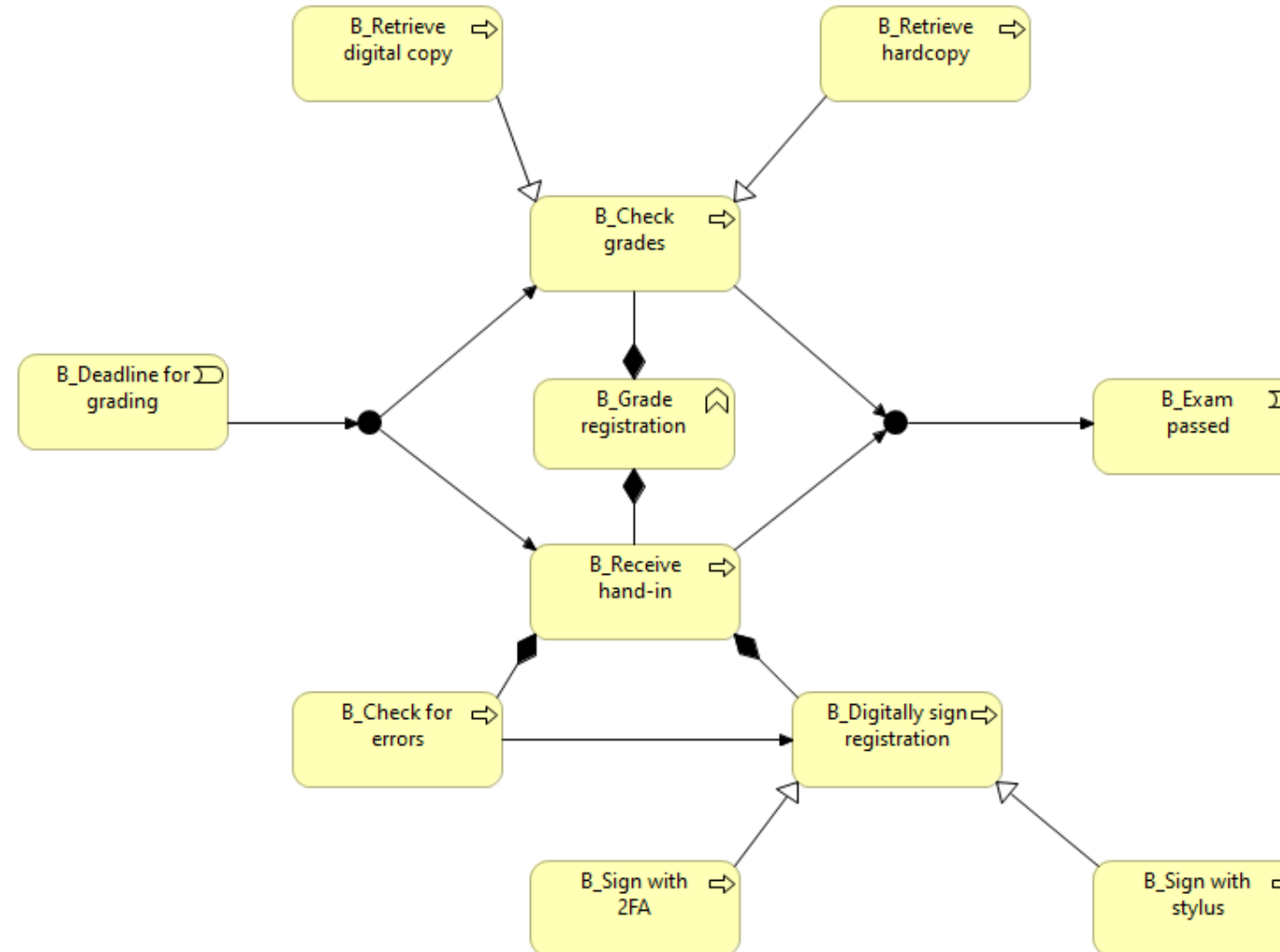


# Passive elements



A concept used within a particular business domain

# Nesting example

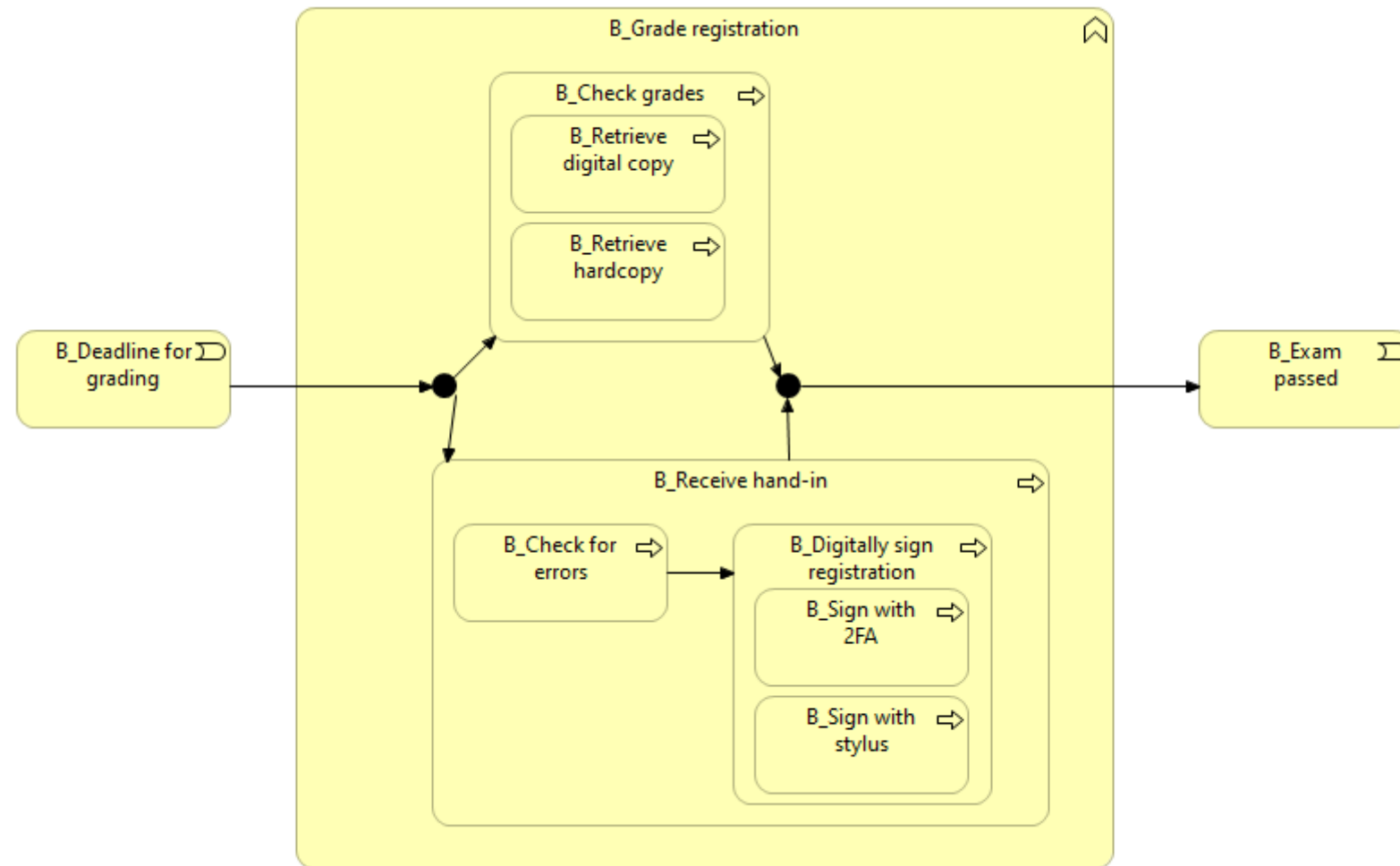


- Each element is shown with all its relations



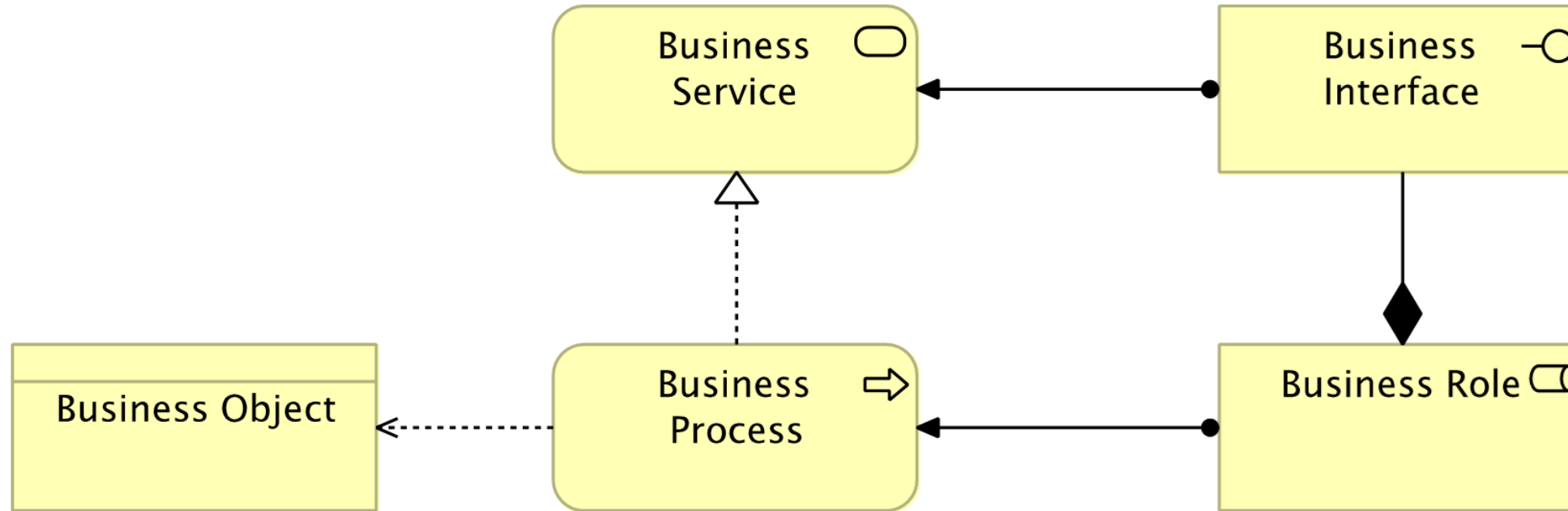


# Nesting example



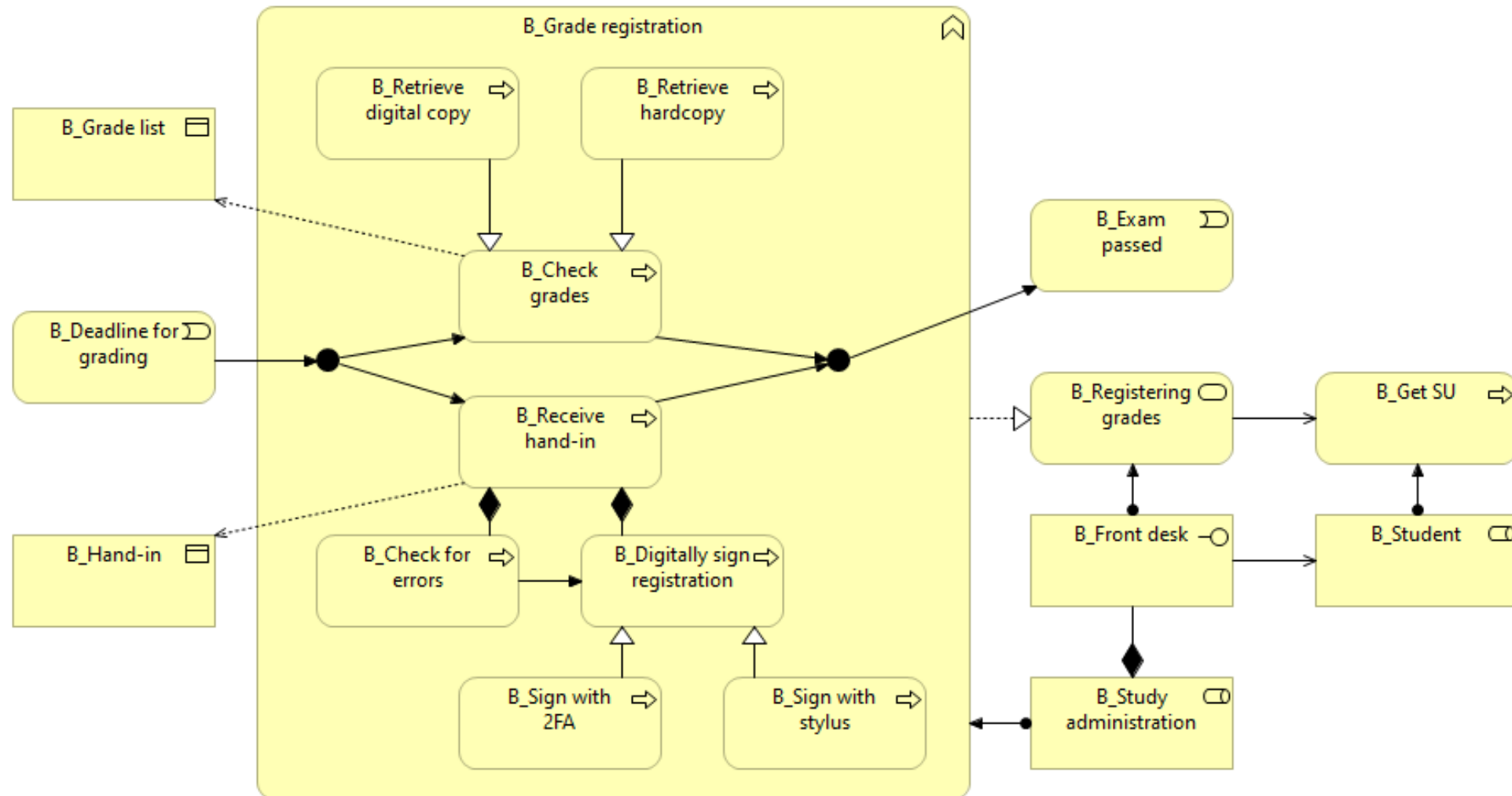
- Doing so, we have a more compact visualization, but we lose details

# Basic business pattern



- ←····· This is the **Access** relation
- ◆—— This is the **Composition** relation
- ▷ This is the **Realization** relation
- ←——● This is the **Assignment** relation

# Basic business pattern: example



# Application layer

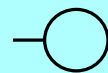
# Active elements

Application  
component



An encapsulation of application functionality aligned to implementation structure, which is modular and replaceable

Application  
interface



A point of access where an application service is made available to a user, another application component, or a node

# Behavior elements

Application  
function

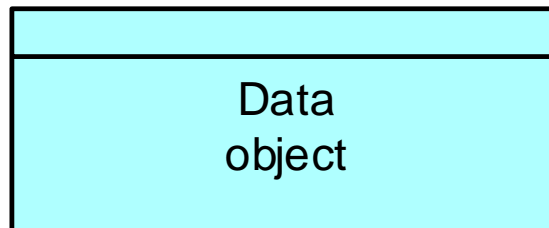


Automated behavior that can be performed by an application component

Application  
service

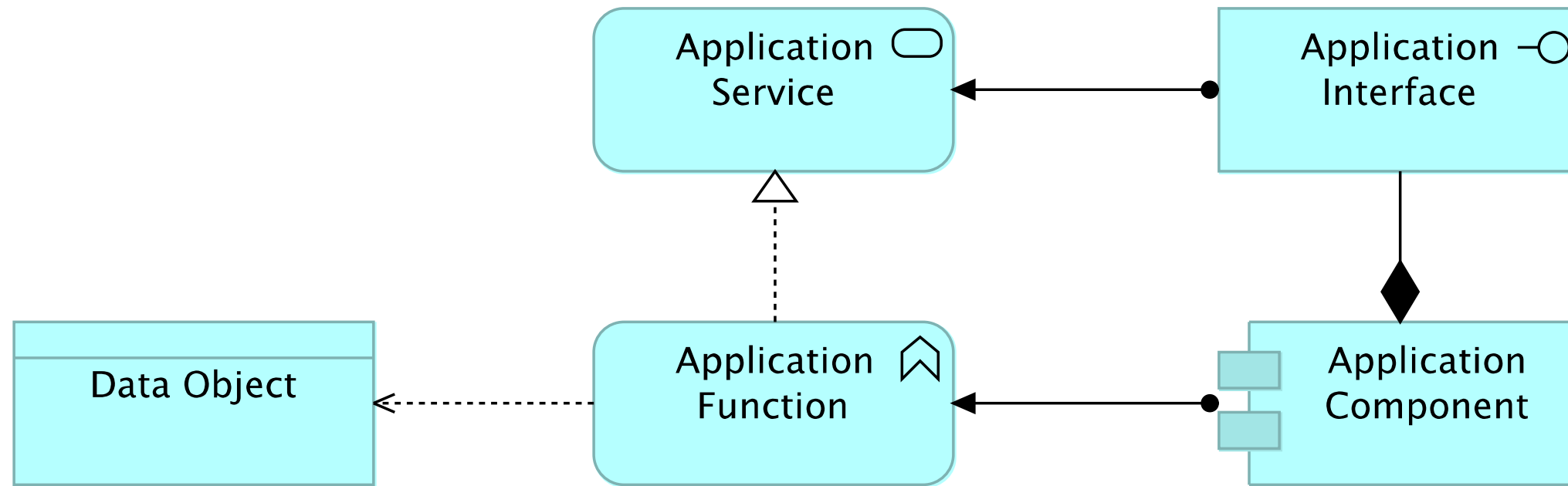
An explicitly defined exposed application behavior

# Passive elements



Data structured for information processing

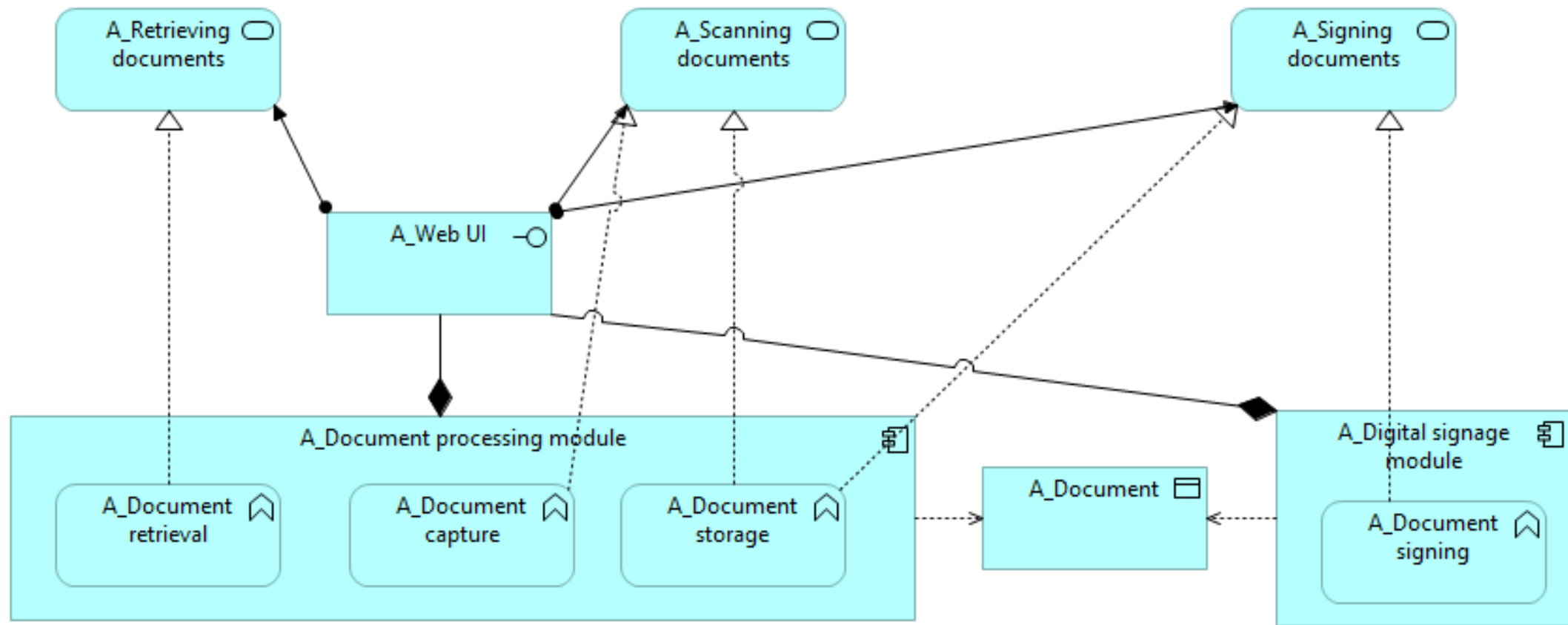
# Basic application pattern

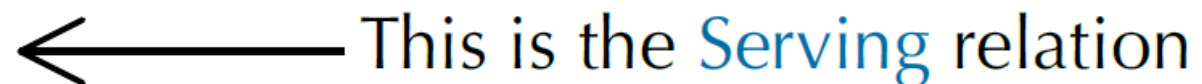


- ←..... This is the **Access** relation
- ◆— This is the **Composition** relation
- △..... This is the **Realization** relation
- ←●— This is the **Assignment** relation

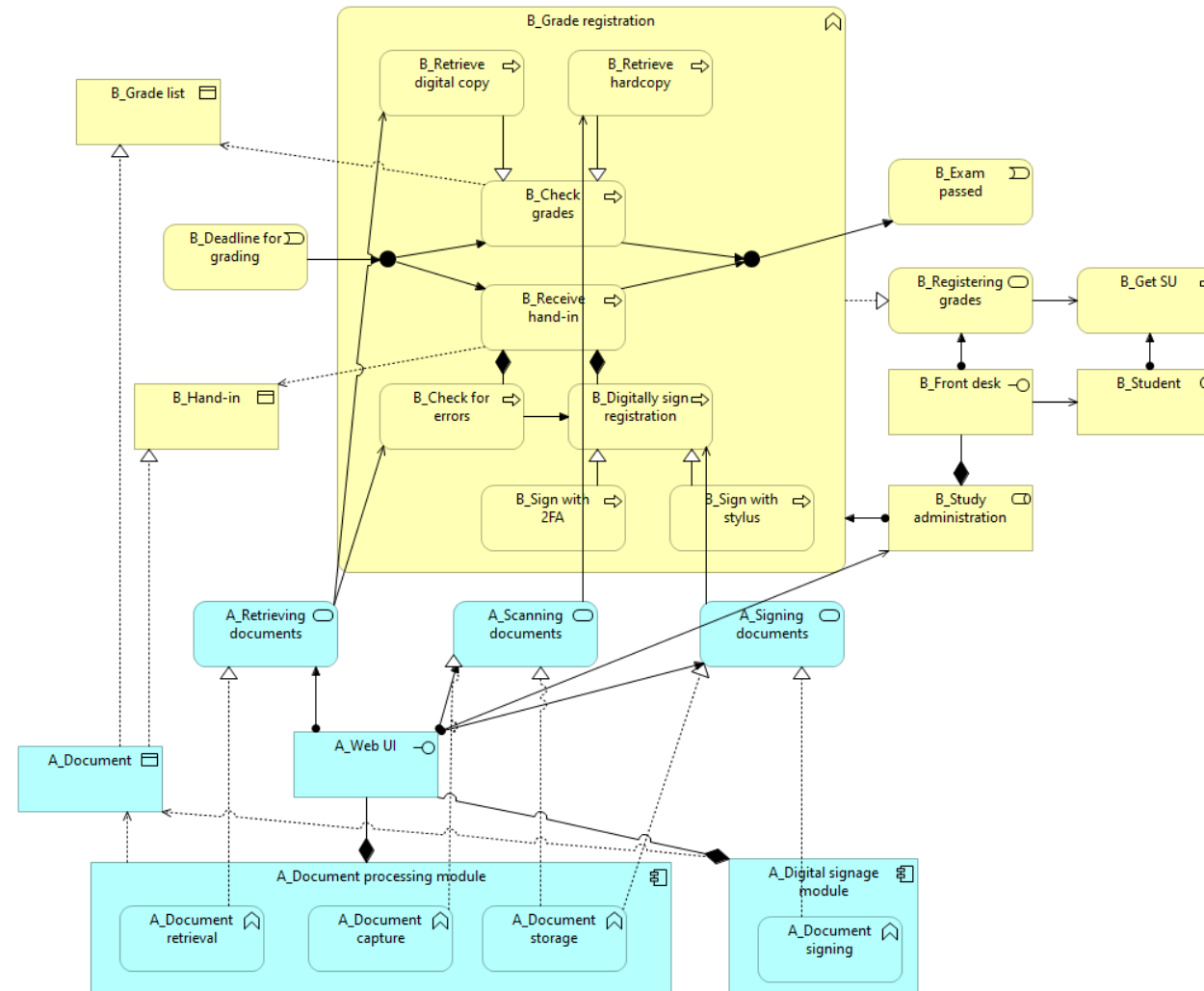


# Basic application pattern: example



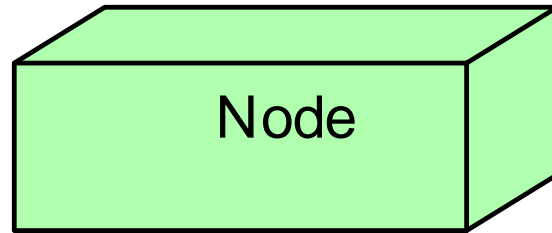


# Business to application layer alignment: example

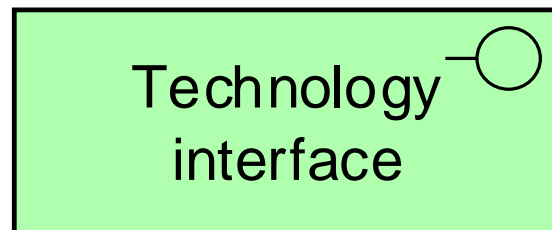


# Technology layer

## Active elements

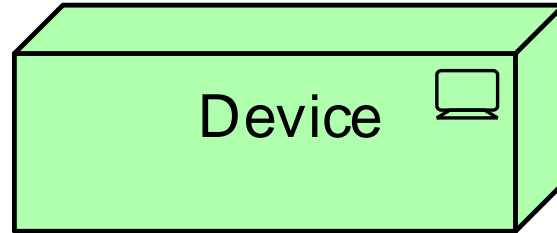


A computational or physical resource that hosts, manipulates or interacts with other computational or physical resources

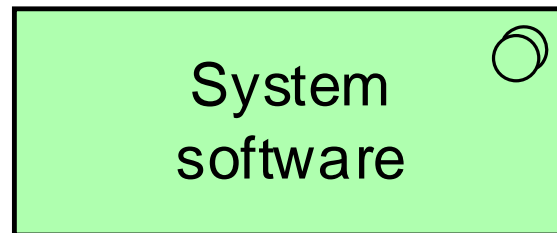


A point of access where a technology service offered by a node is made accessible

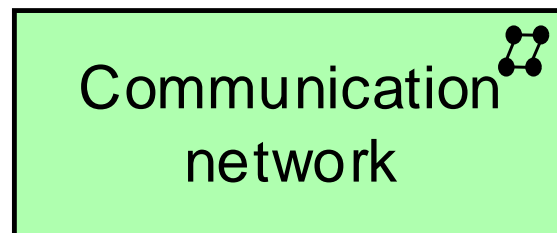
# Active elements



A physical IT resource upon which system software and artifacts may be deployed

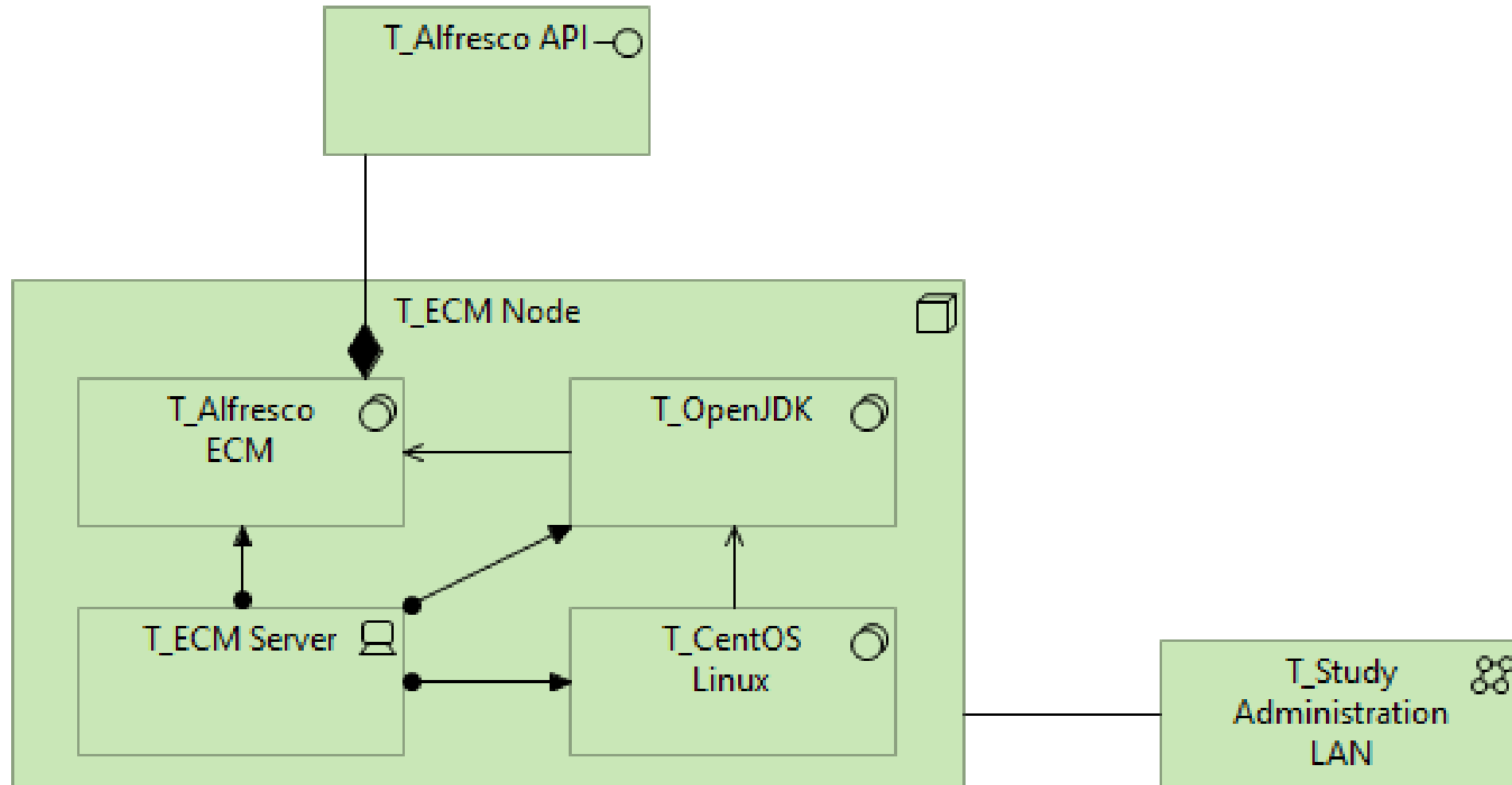


Software that provides or contributes to an environment for storing, executing and using software and data deployed within it

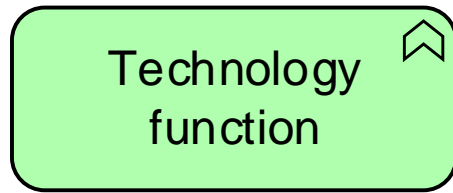


A set of structures that connects computer systems or other devices for transmission, routing and reception of data

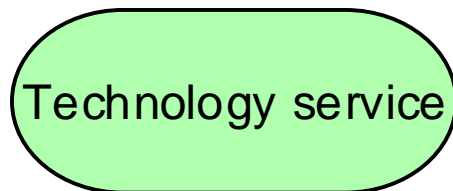
# Active elements: example



# Behavior elements



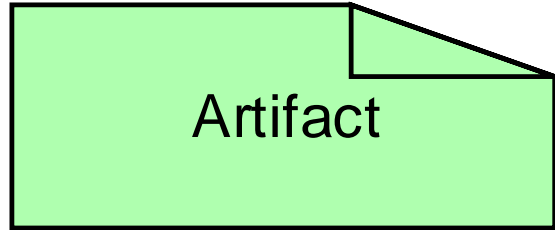
A behavior element that groups infrastructural behavior that can be performed by a node



Externally visible unit of functionality, provided by one or more nodes, exposed through well-defined interfaces

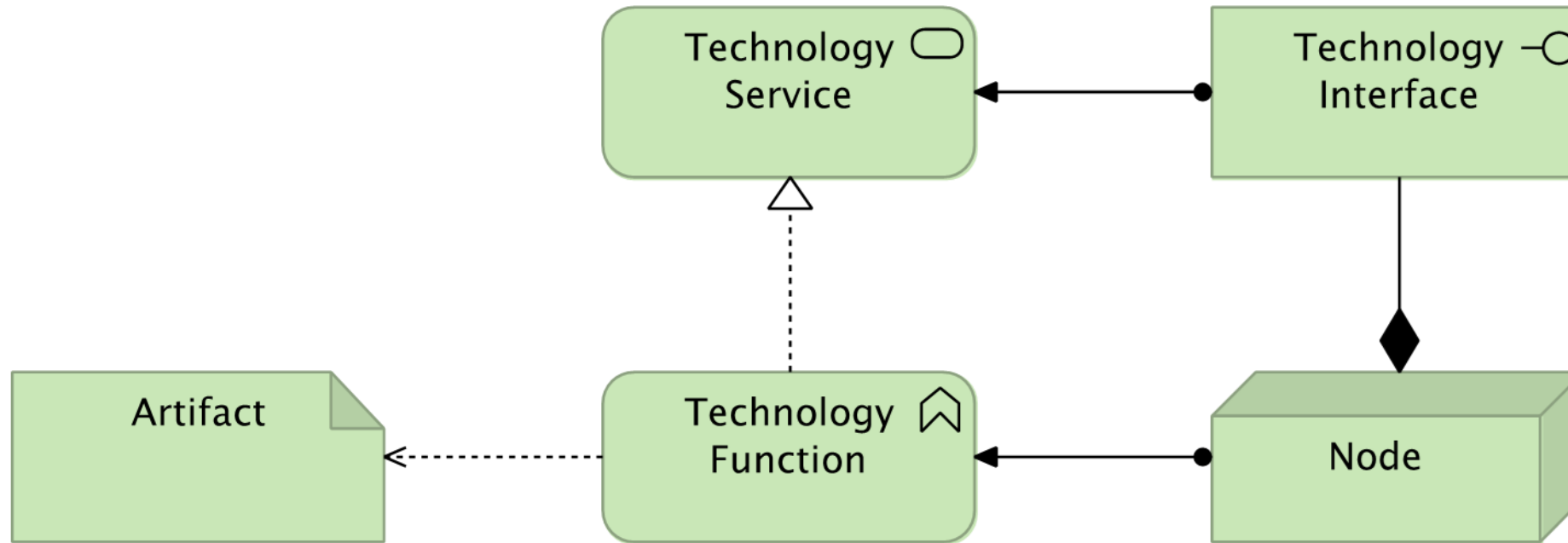


# Passive elements



A piece of data that is used or produced in a software development process or by deployment and operation of a system

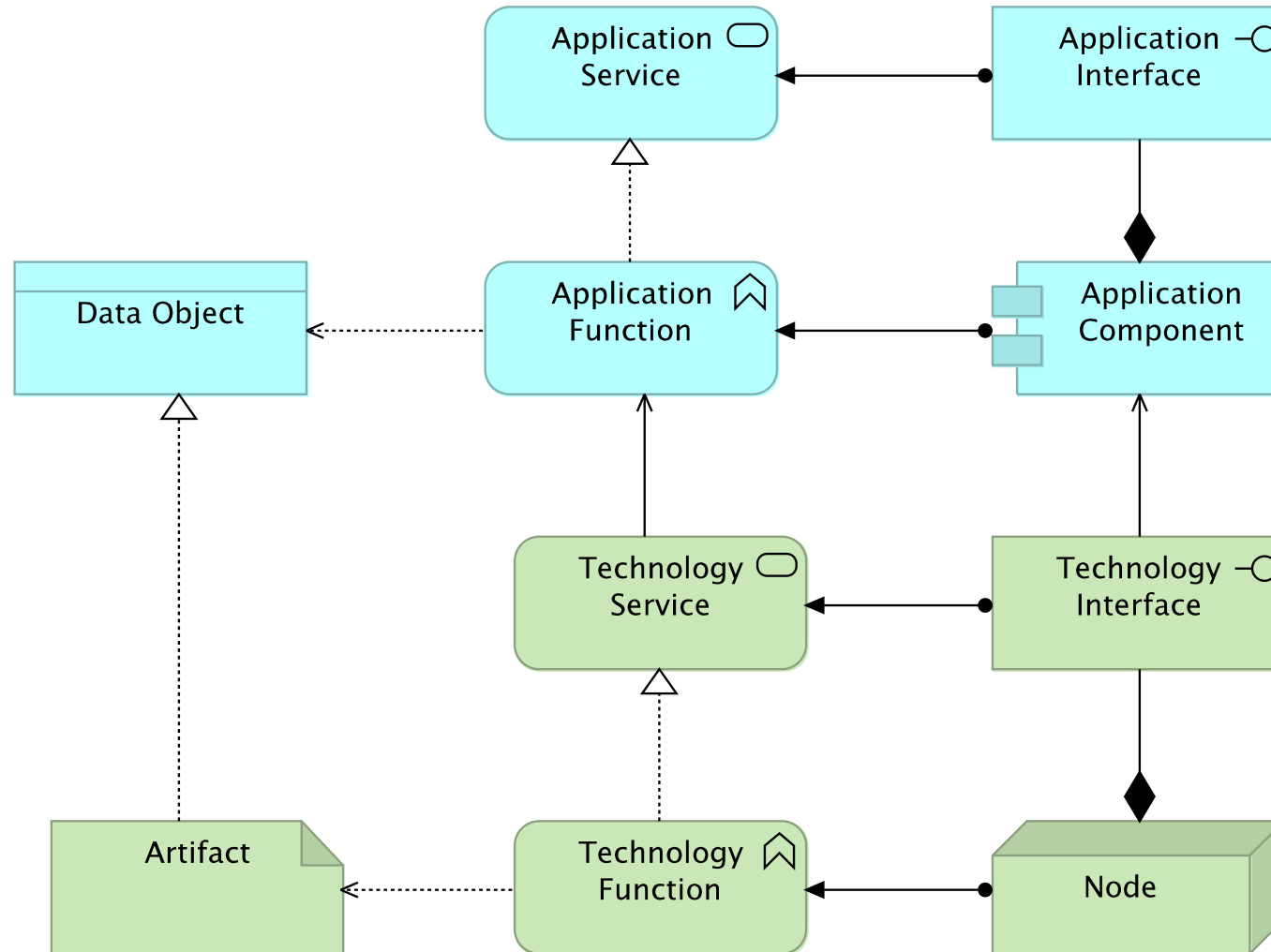
# Basic technology pattern



- ←····· This is the **Access** relation
- ◆— This is the **Composition** relation
- △····· This is the **Realization** relation
- ←—● This is the **Assignment** relation

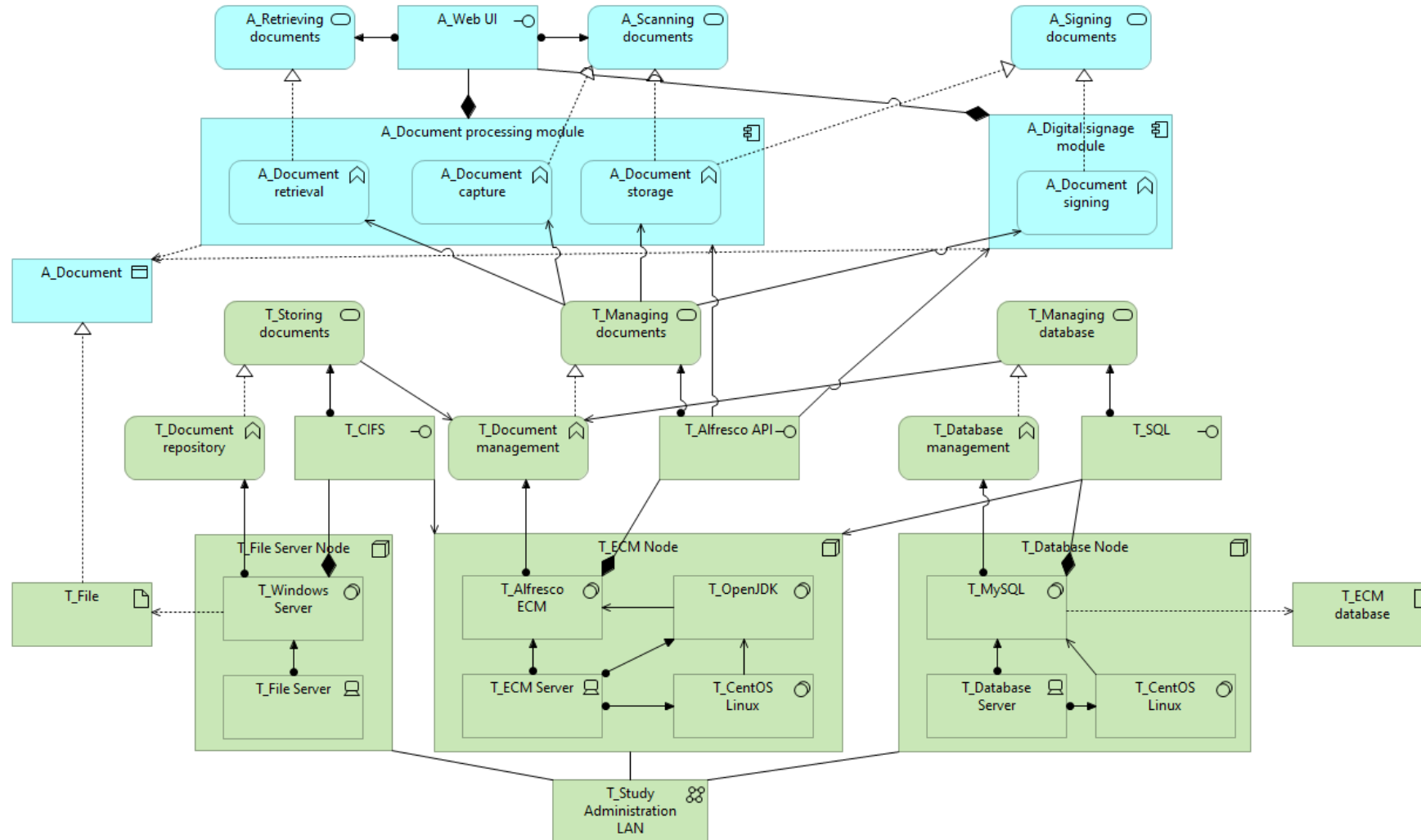


# Application to technology alignment

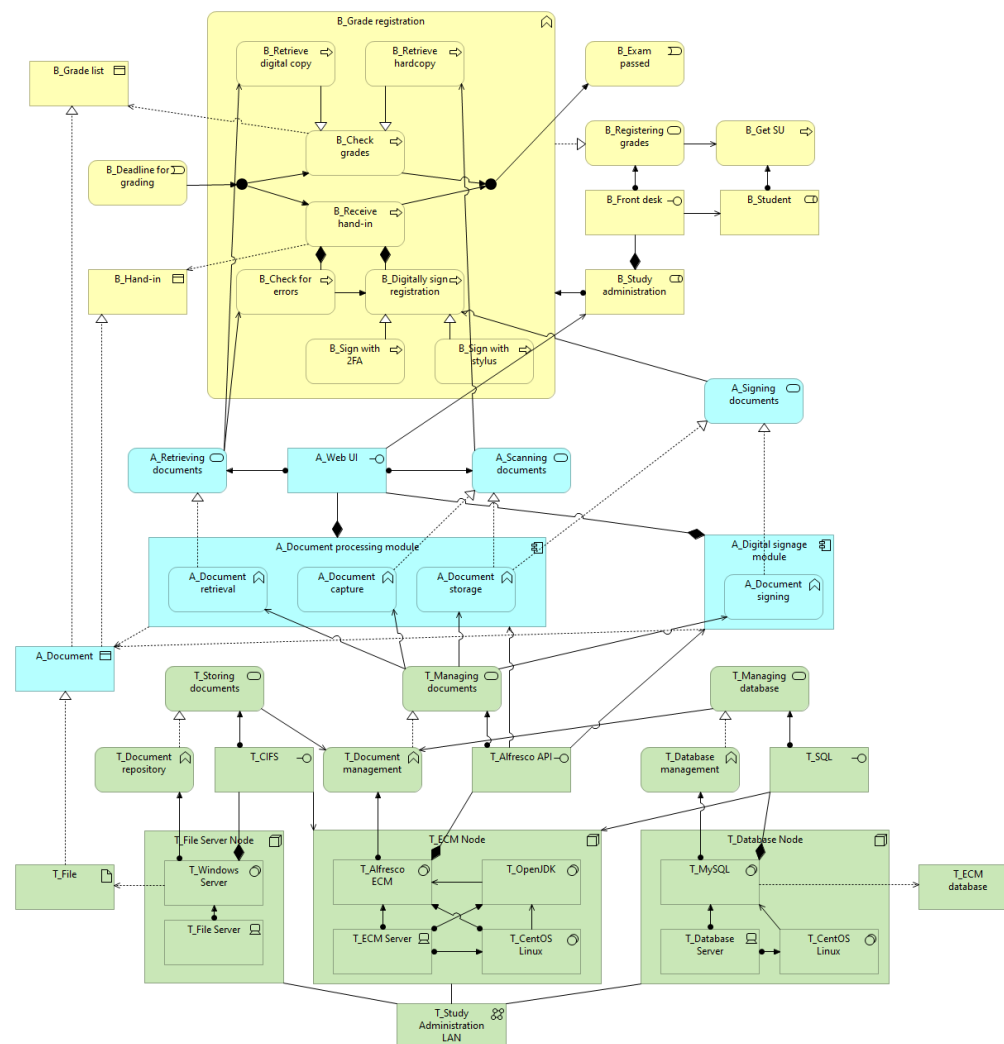


← This is the **Serving** relation

# Application to technology alignment: example



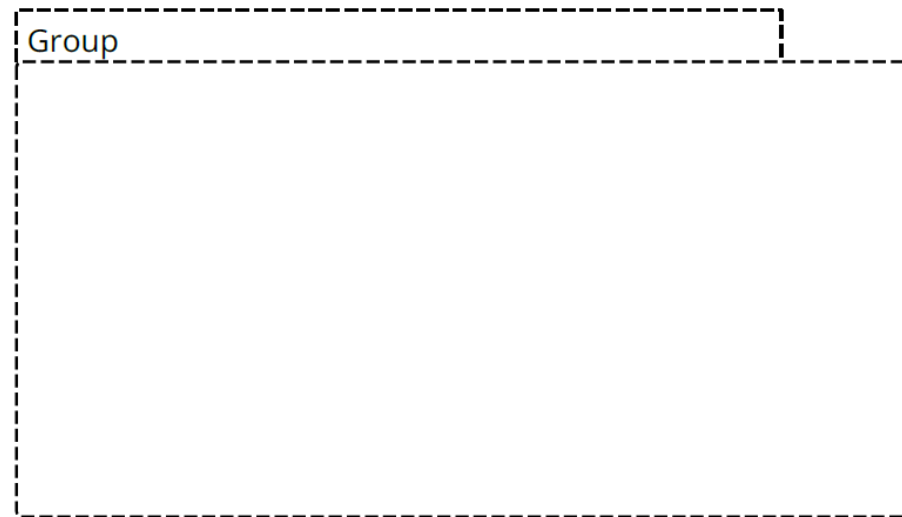
# Complete example



# Composite elements

# Grouping

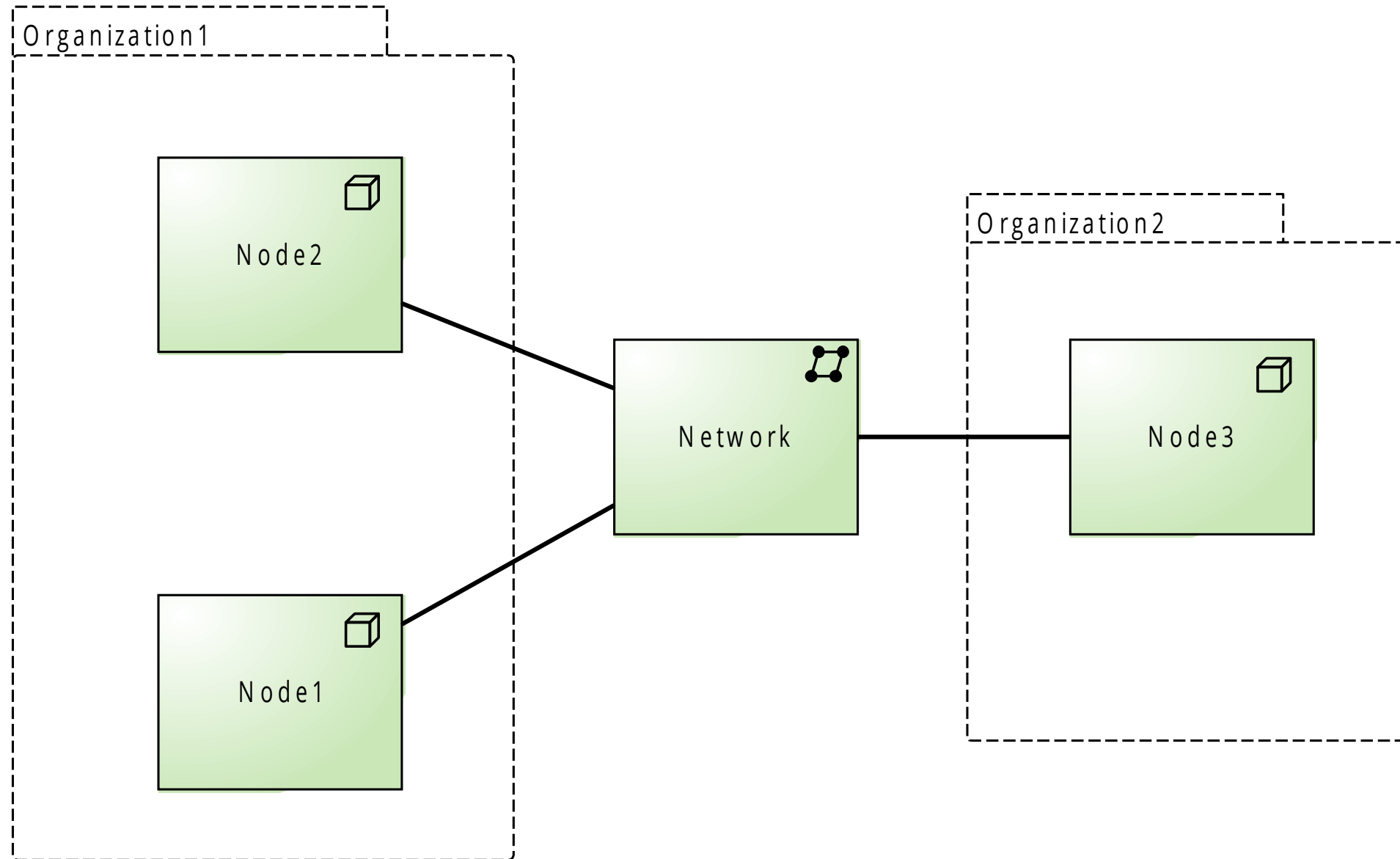
- Aggregate elements together
- Aggregate elements of the same (external) organization






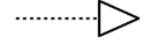




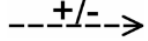

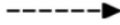




- We group external organizations, while we usually not group the target organization for clarity



# Grouping example: two different organizations

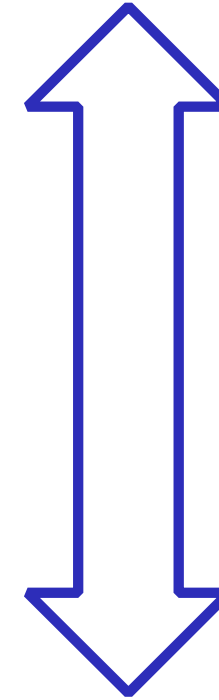


# Relationships

Structural Relationships		Notation
Composition	Indicates that an element consists of one or more other concepts.	
Aggregation	Indicates that an element groups a number of other concepts.	
Assignment	Expresses the allocation of responsibility, performance of behavior, or execution.	
Realization	Indicates that an entity plays a critical role in the creation, achievement, sustenance, or operation of a more abstract entity.	
Dependency Relationships		Notation
Serving	Models that an element provides its functionality to another element.	
Access	Models the ability of behavior and active structure elements to observe or act upon passive structure elements.	  
Influence	Models that an element affects the implementation or achievement of some motivation element.	
Dynamic Relationships		Notation
Triggering	Describes a temporal or causal relationship between elements.	
Flow	Transfer from one element to another.	
Other Relationships		Notation
Specialization	Indicates that an element is a particular kind of another element.	
Association	Models an unspecified relationship, or one that is not represented by another ArchiMate relationship.	
Junction	Used to connect relationships of the same type.	 (And) Junction  Or Junction

## Relationships strength

- Access
- Serving
- Realization
- Assignment
- Aggregation
- Composition

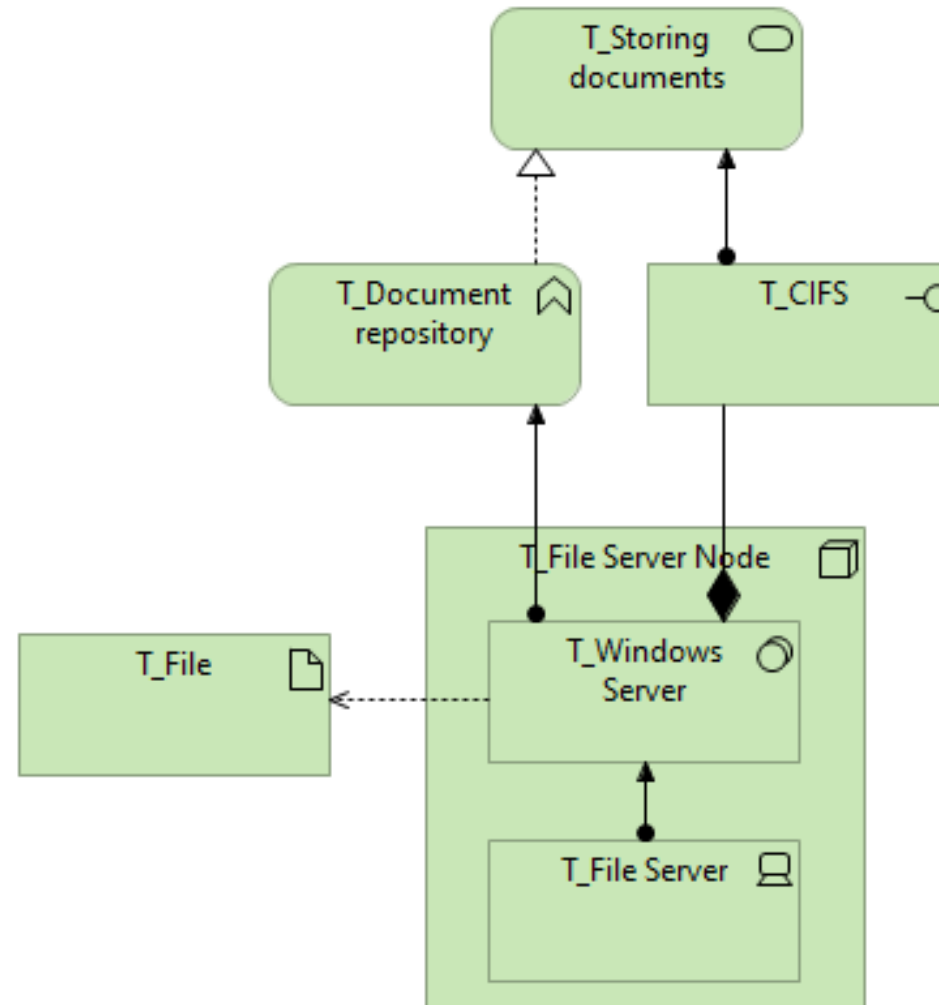


Weaker

Association relation: generic relation

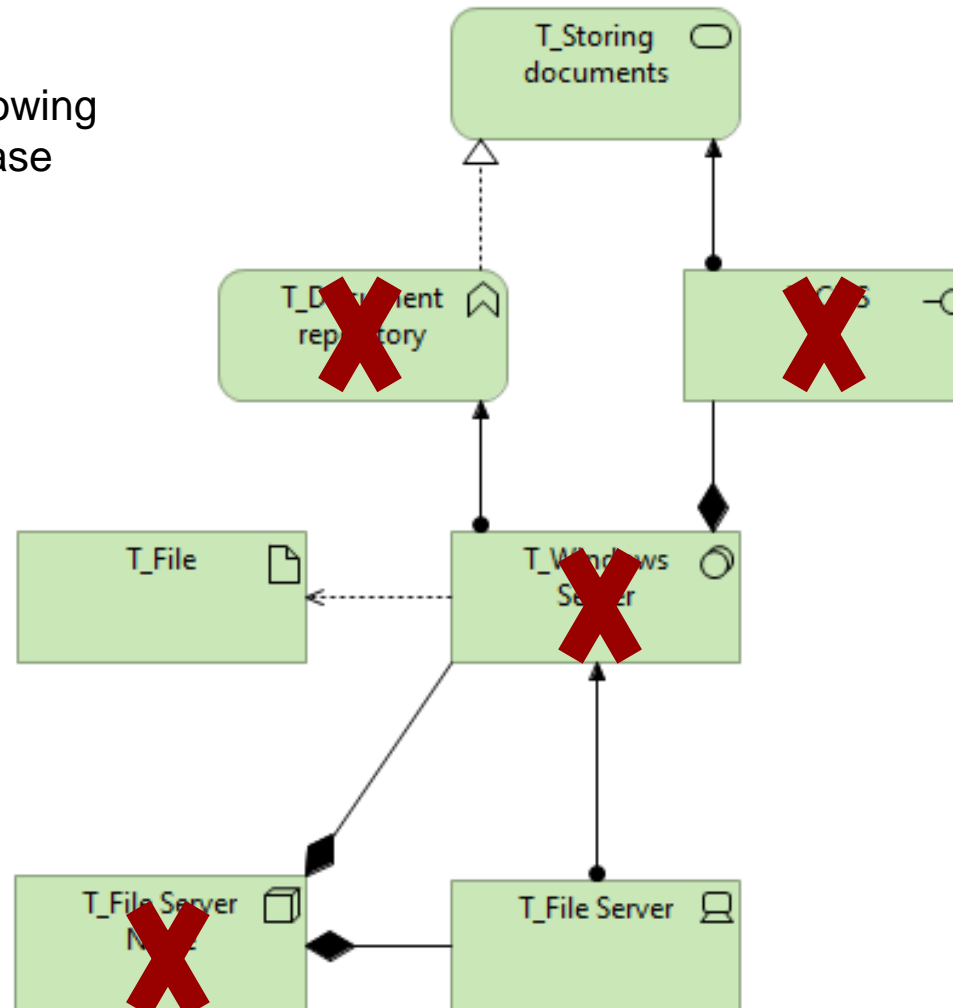
Stronger

# Derived Relationships: example



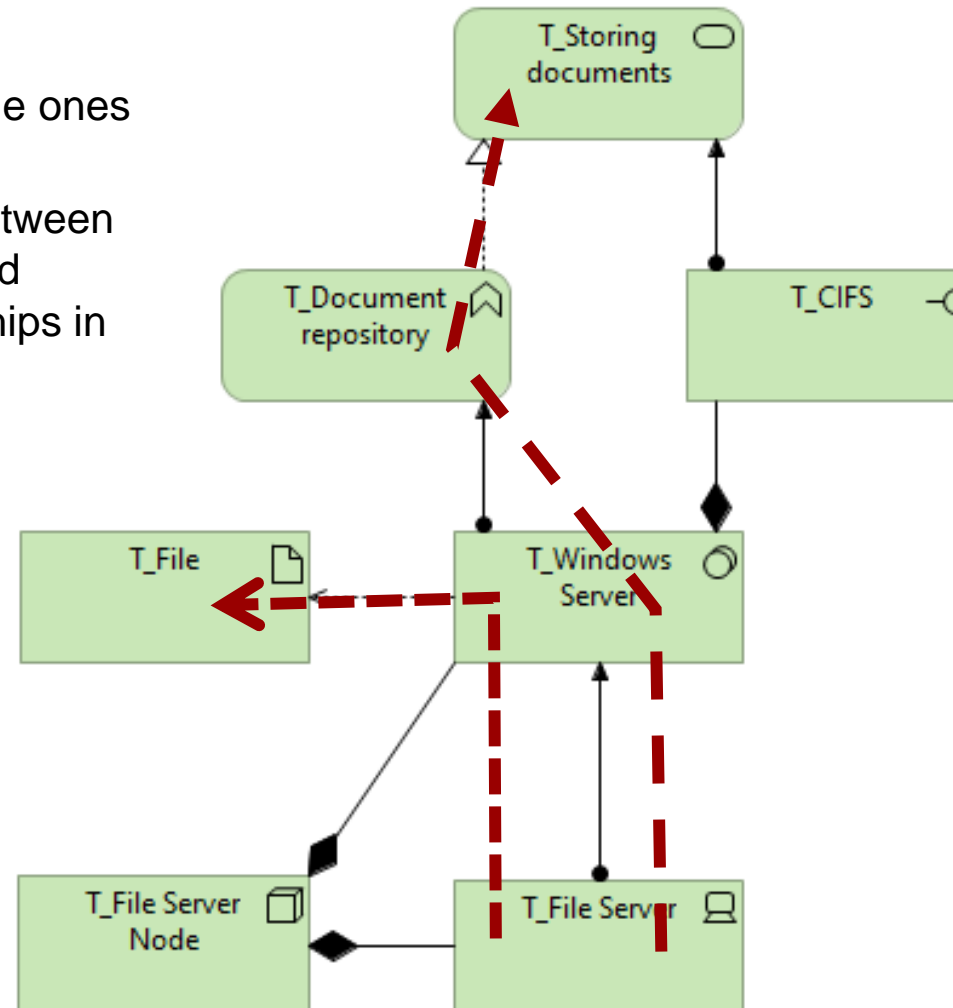
# Derived Relationships: example

- We want to abstract from details, showing only elements relevant for our use case



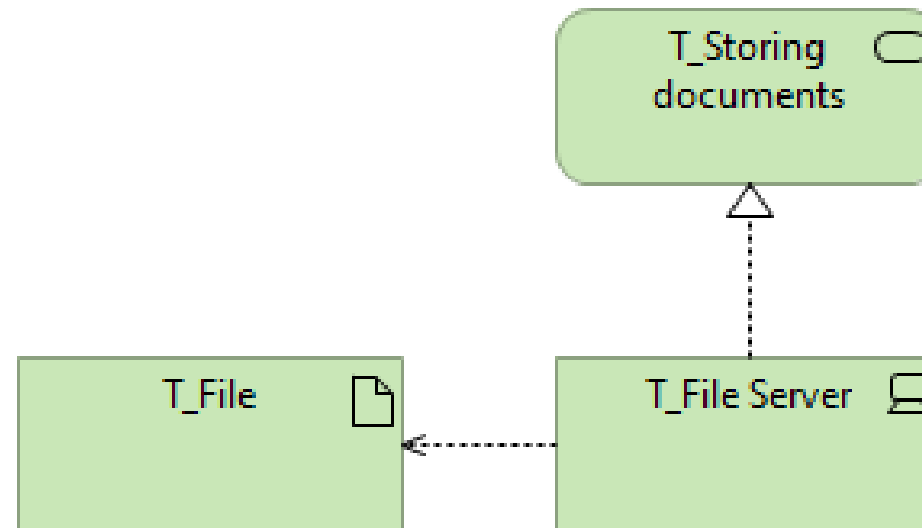
# Derived Relationships: example

- We can derive relations, based on the ones in the detailed model.
- Intuitively, there must exist a path between two elements, and the type of derived relationship is the weakest relationships in the path



# Derived Relationships: example

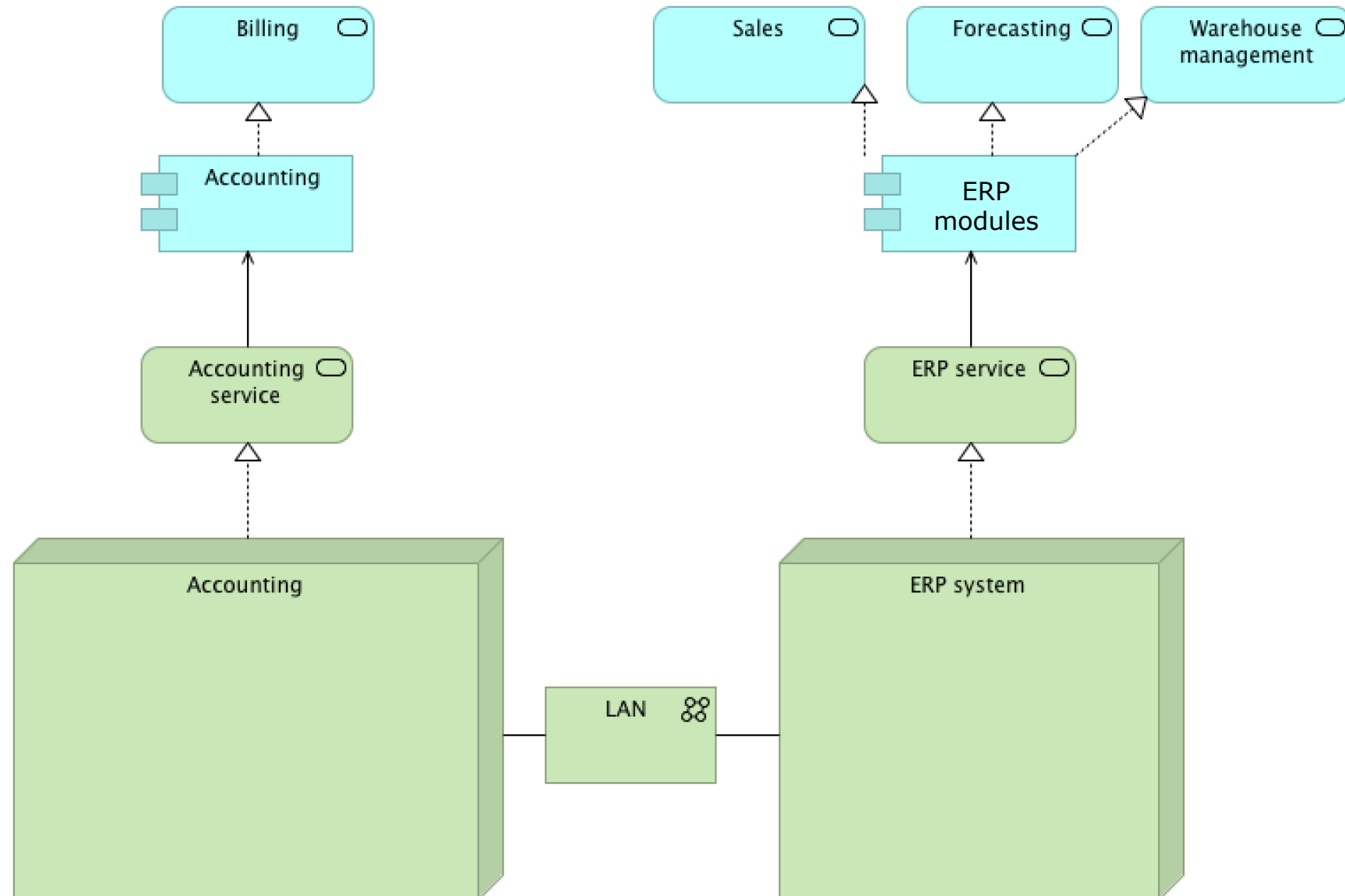
- The resulting simplified diagram still explains the relationships between elements



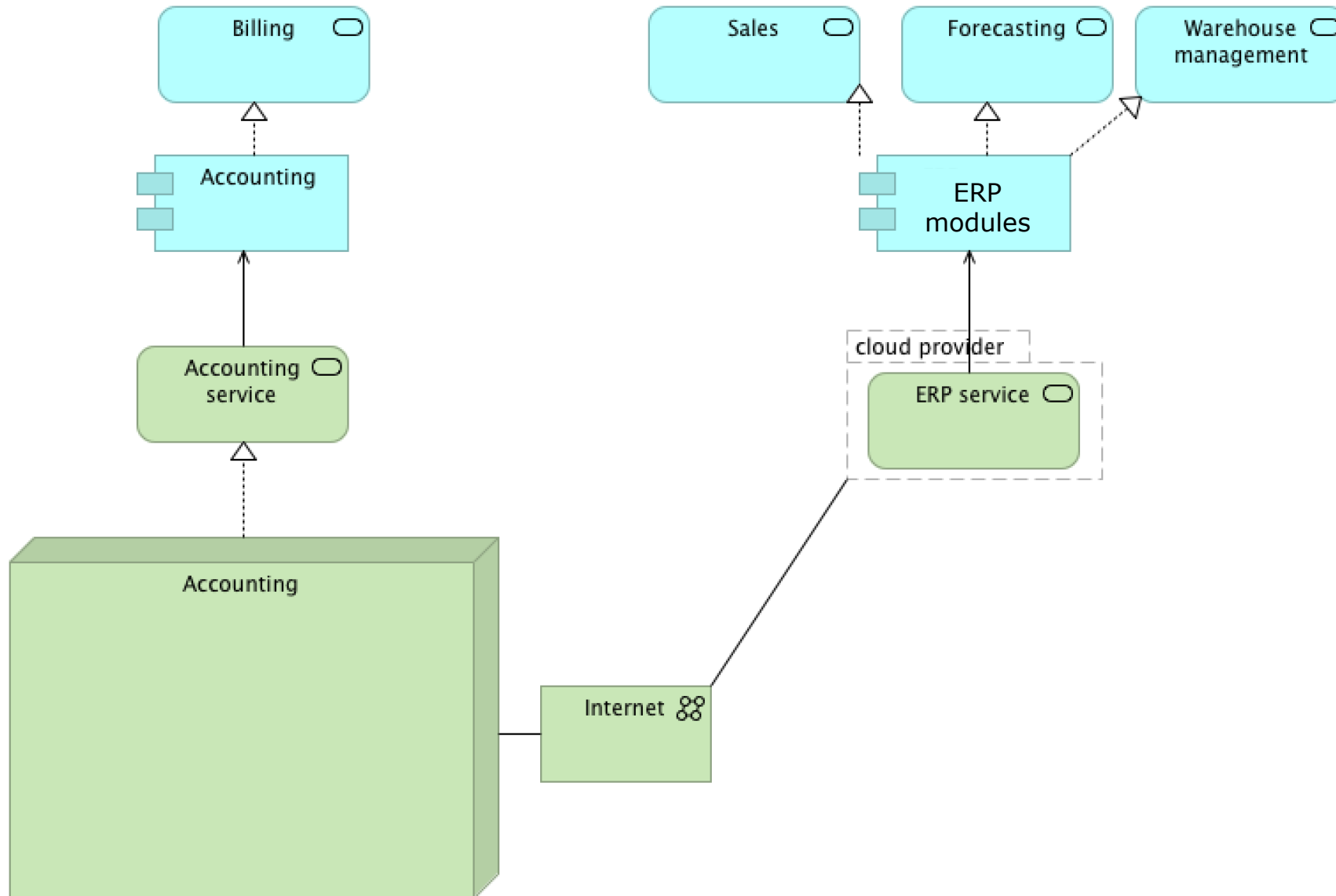
# Useful patterns



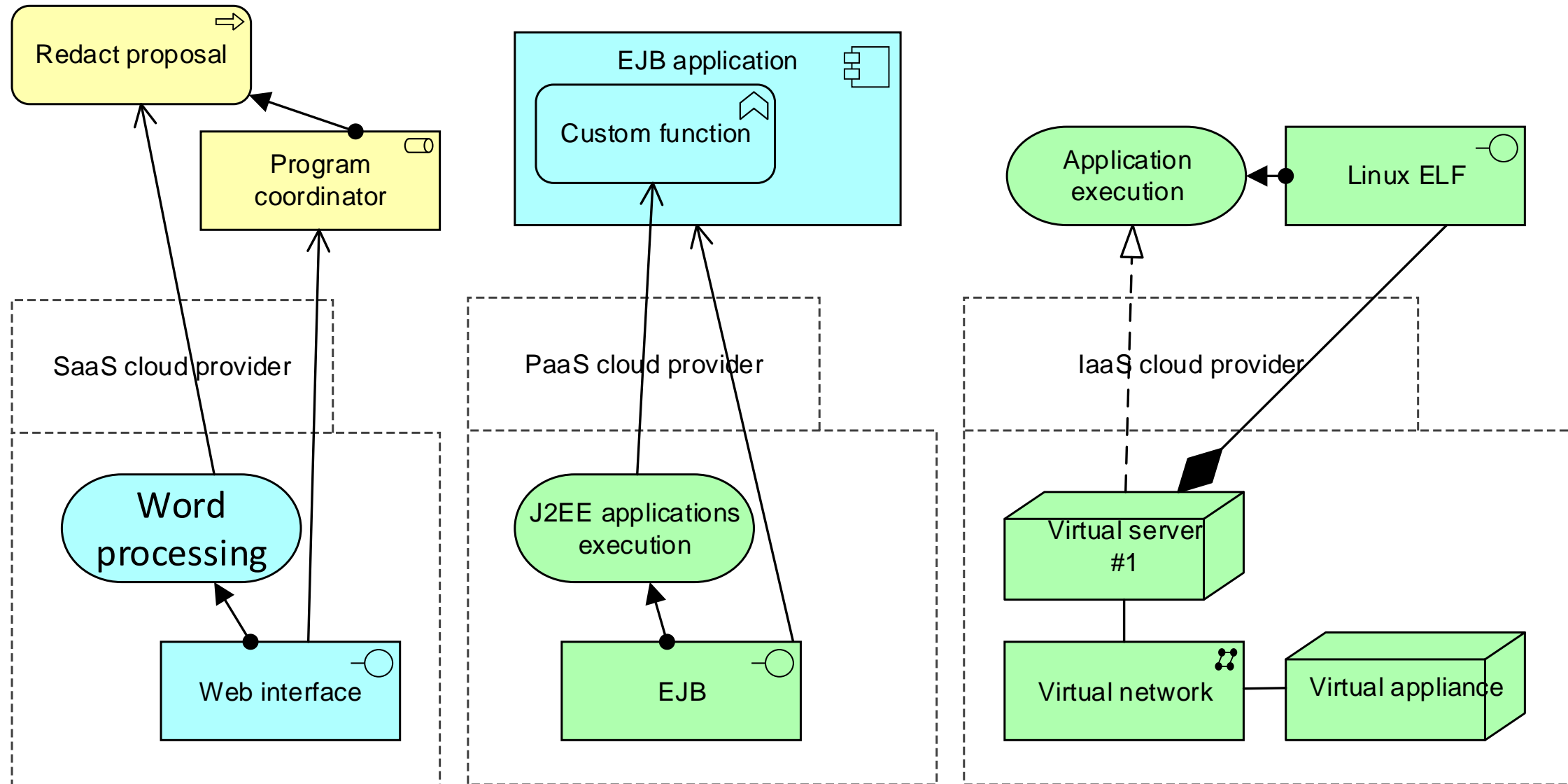
# On-premise backend



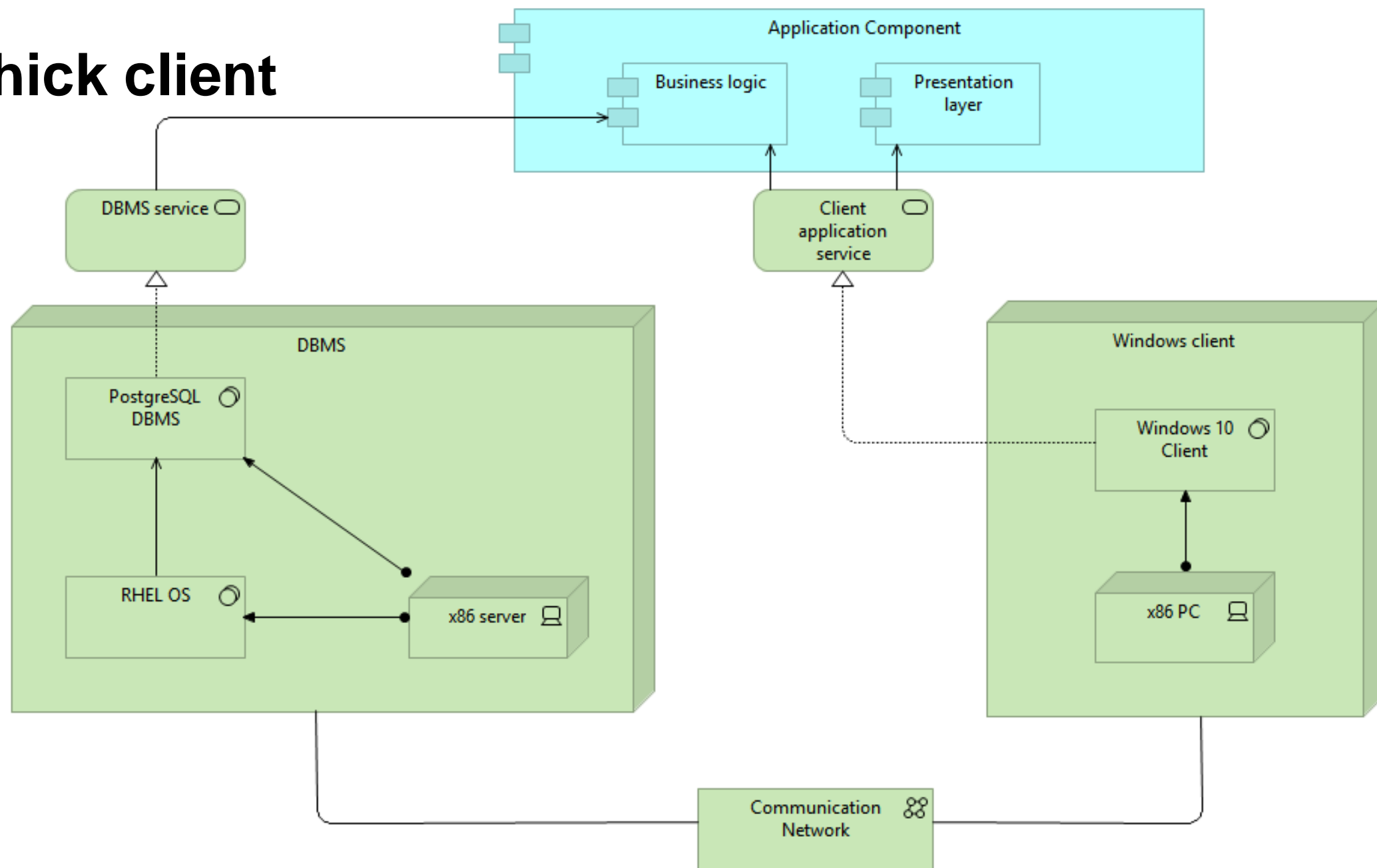
# Switching to a cloud-based technology service



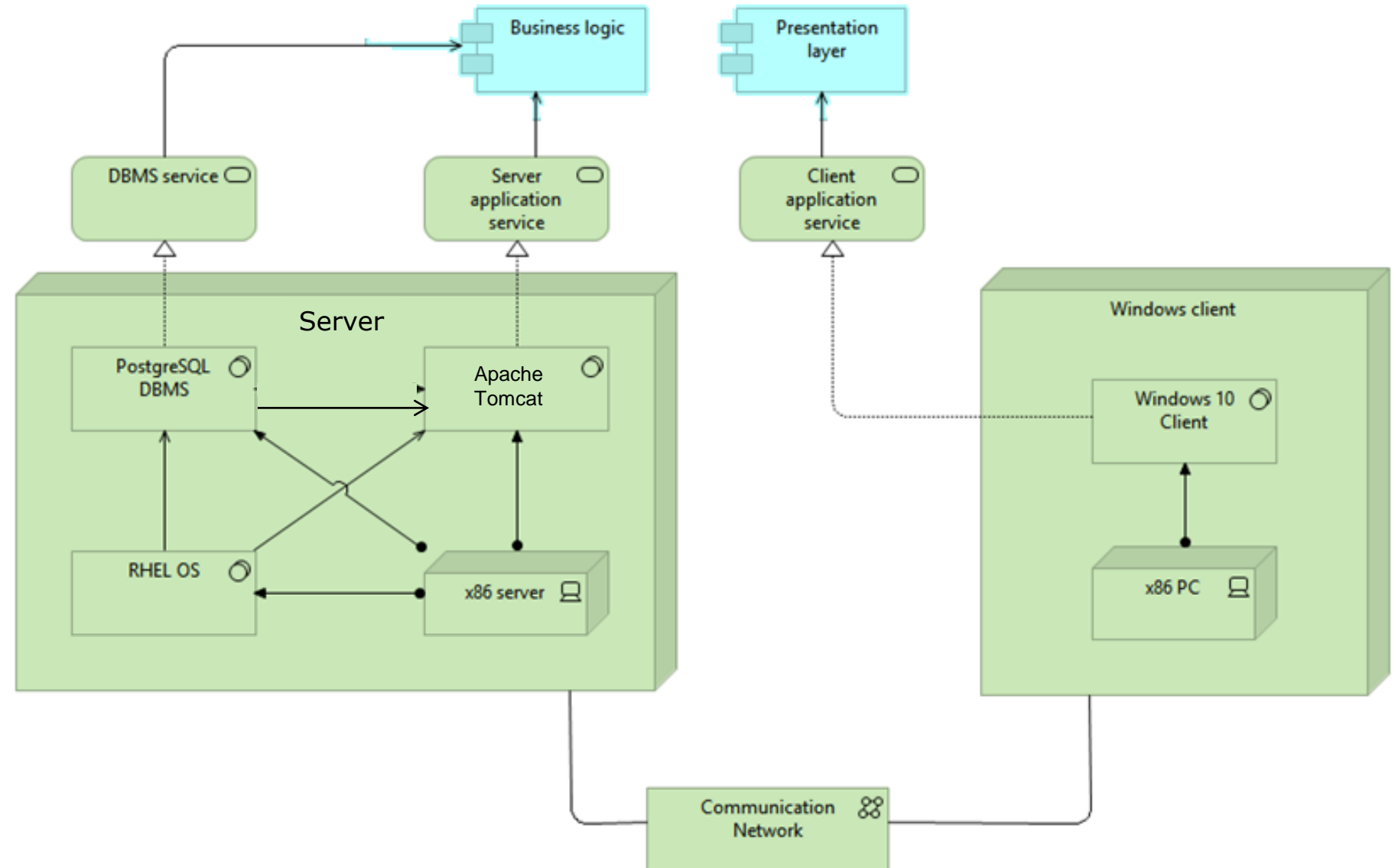
# Cloud levels



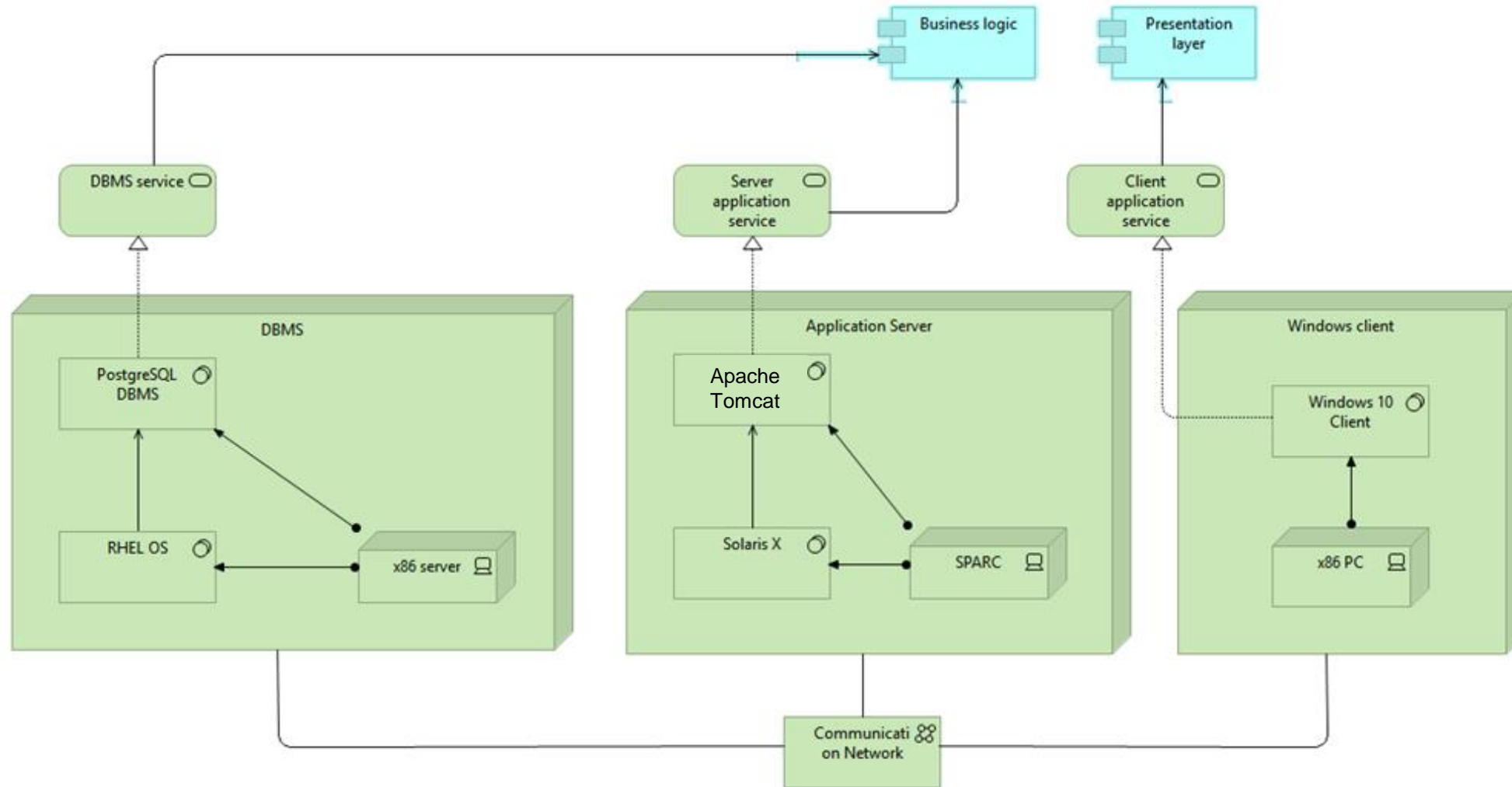
# 2-tier thick client



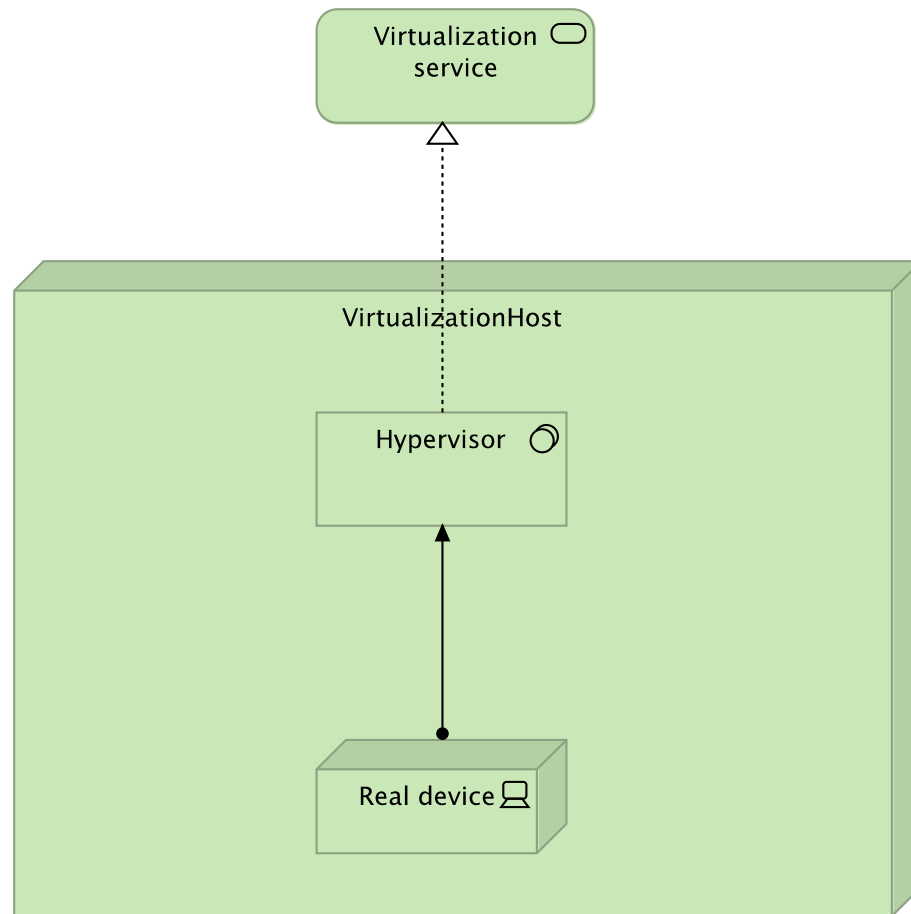
# Switching to a 2-tier thin client



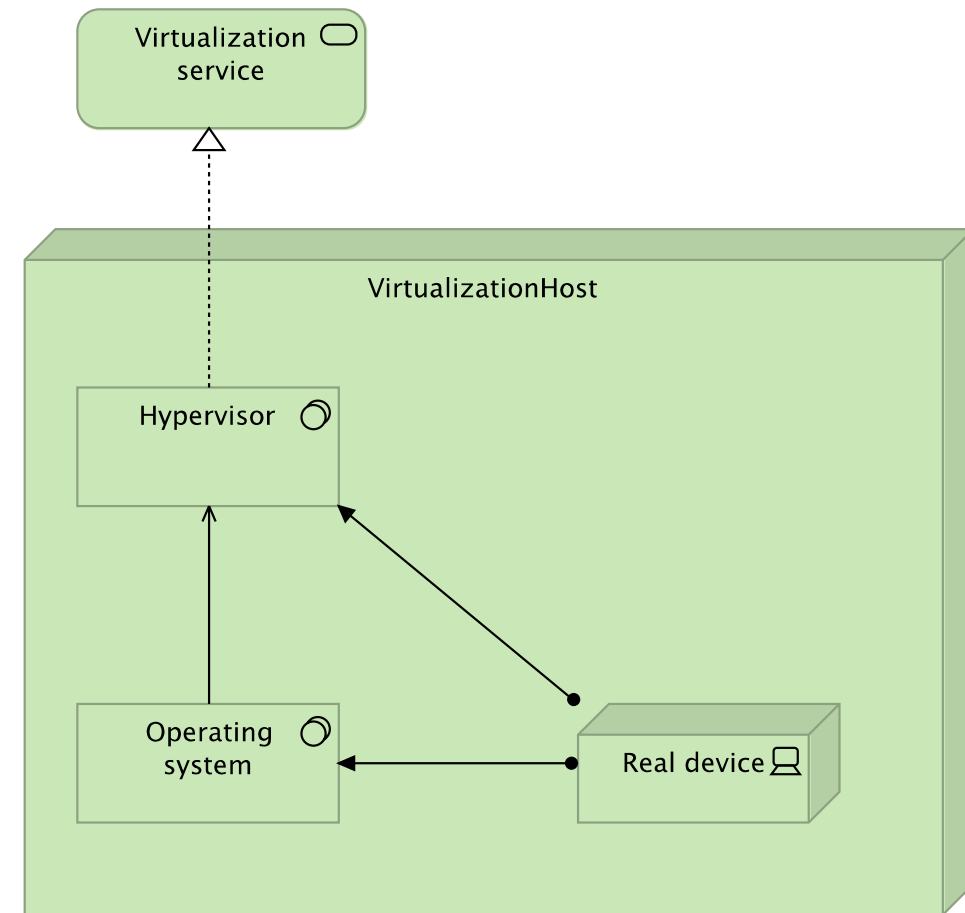
# Switching to a 3-tier architecture



# Bare metal vs hosted virtualization

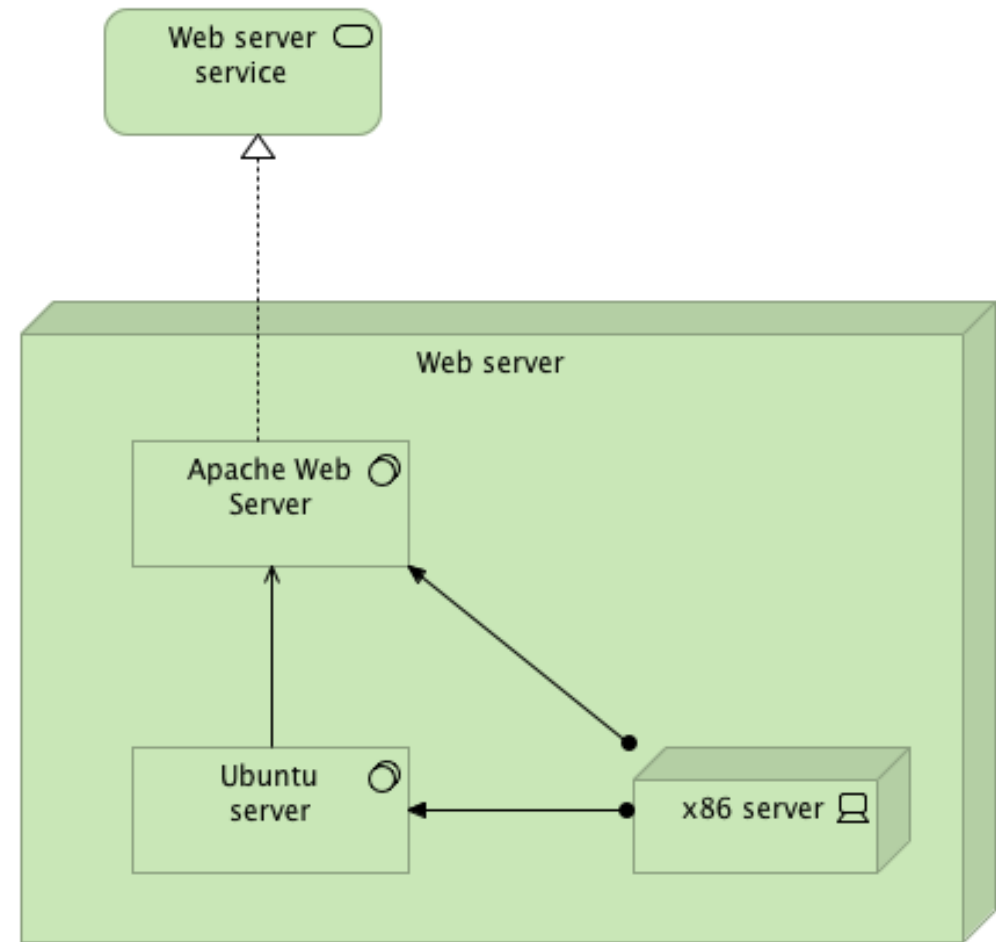
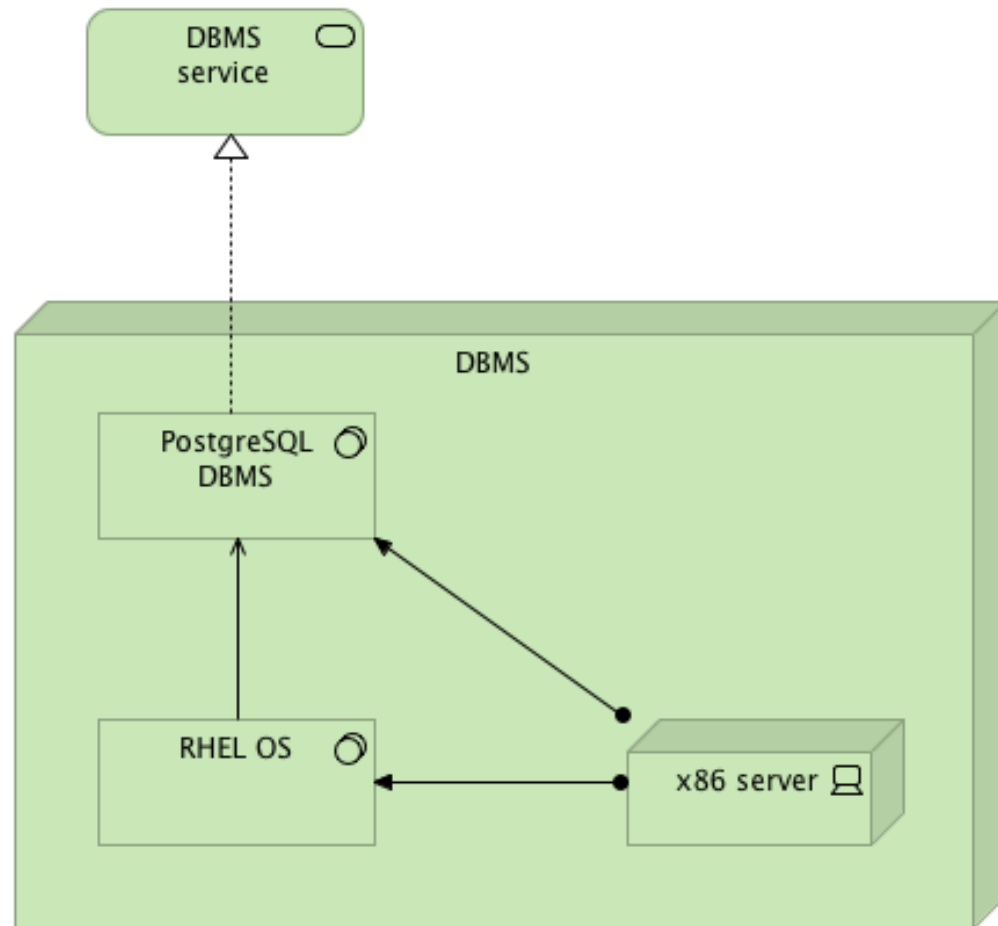


- Bare metal



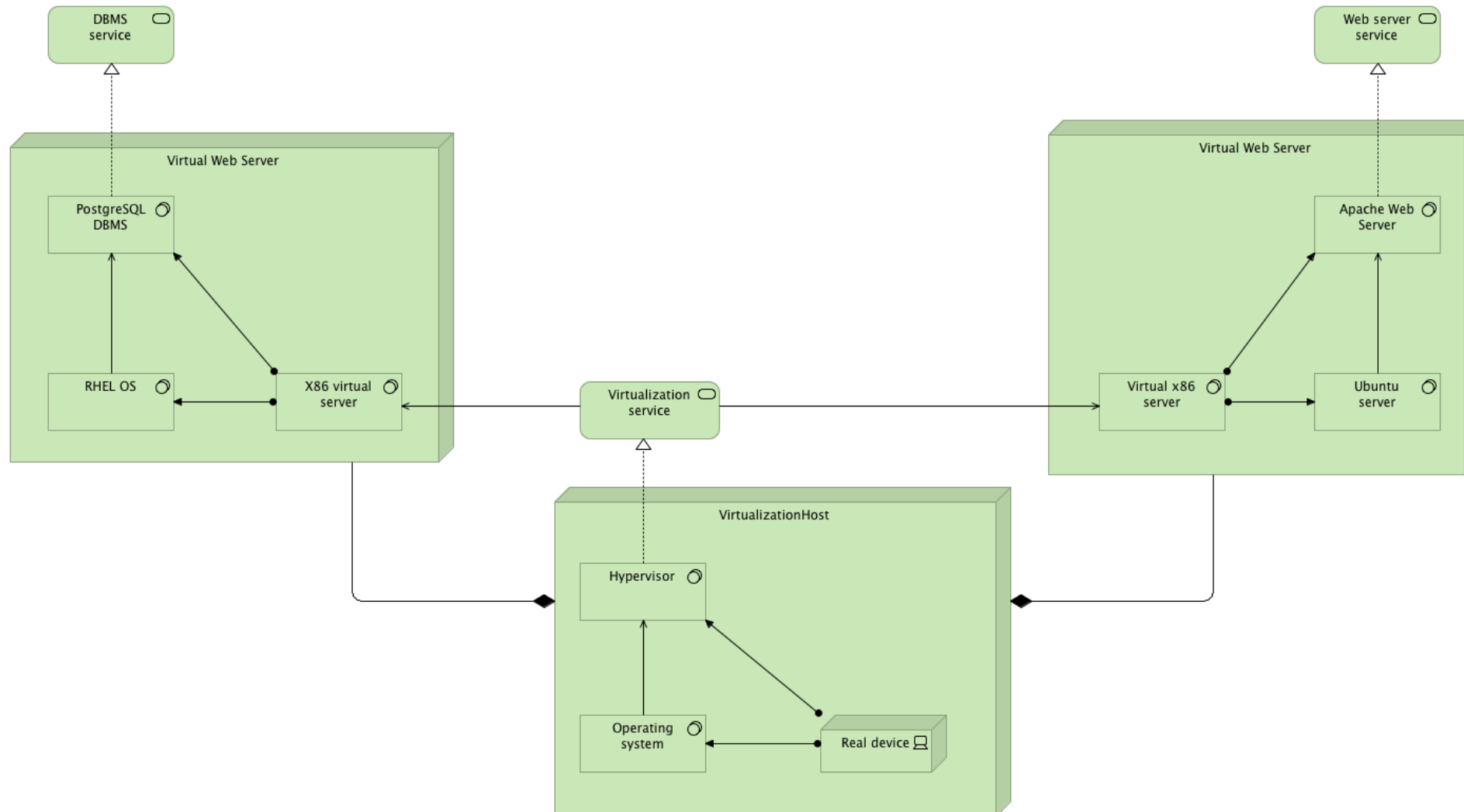
- Hosted

# Physical servers





# Virtualized servers



# Study material

- Books and articles:
  - Lankhorst et al. - Enterprise Architectures at Work (4th Edition)
  - Available at: <https://link.springer.com/book/10.1007/978-3-662-53933-0>
  - Chapter 5.1 to 5.5, 5.8 to 5.10, 5.13
  - ArchiMate specification
  - Available at: <https://pubs.opengroup.org/architecture/archimate31-doc/>
- Modeling tools:
  - Archi: <https://www.archimatetool.com/>
  - (alternatively) SAP Signavio: <https://academic.signavio.com/p/login>

# Exercises

Please answer all exercises to demonstrate your skills.  
Solutions will be available at 11:45

# Exercise 1 – Speedy

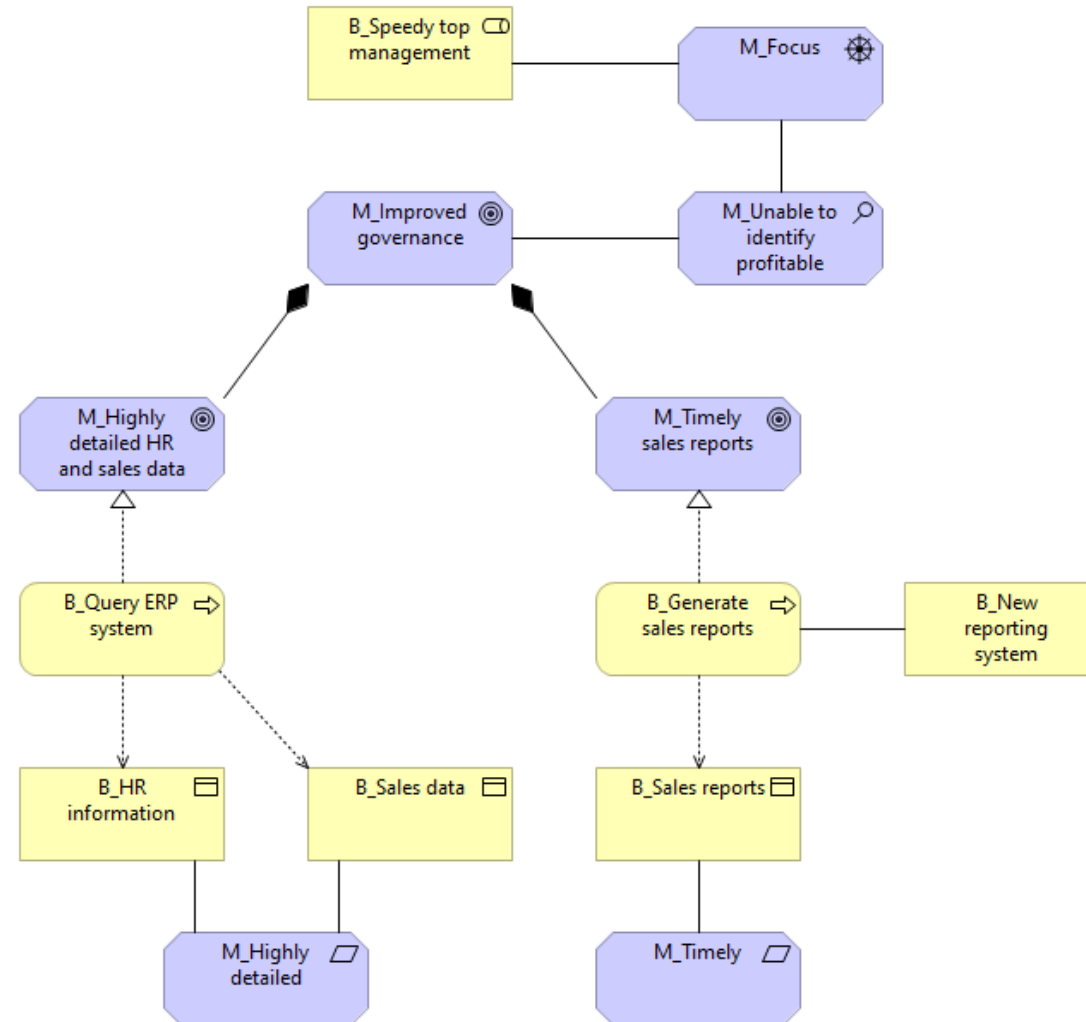
**Speedy** is a delivery company that wants to create a new reporting system for the top management. After inspecting the sales reports, the top management may also need to query the existing ERP system, based on Oracle Fusion, to get detailed sales and HR information.

To implement the reporting system, a data warehouse (DW) based on Microsoft SQL Server 2022 will be used. The DW will run three components responsible for extracting sales data from the ERP database, managing analysis data, and generating sales reports. The DW will run on-premise on a different node than the one running the ERP. Both nodes will share the same LAN.

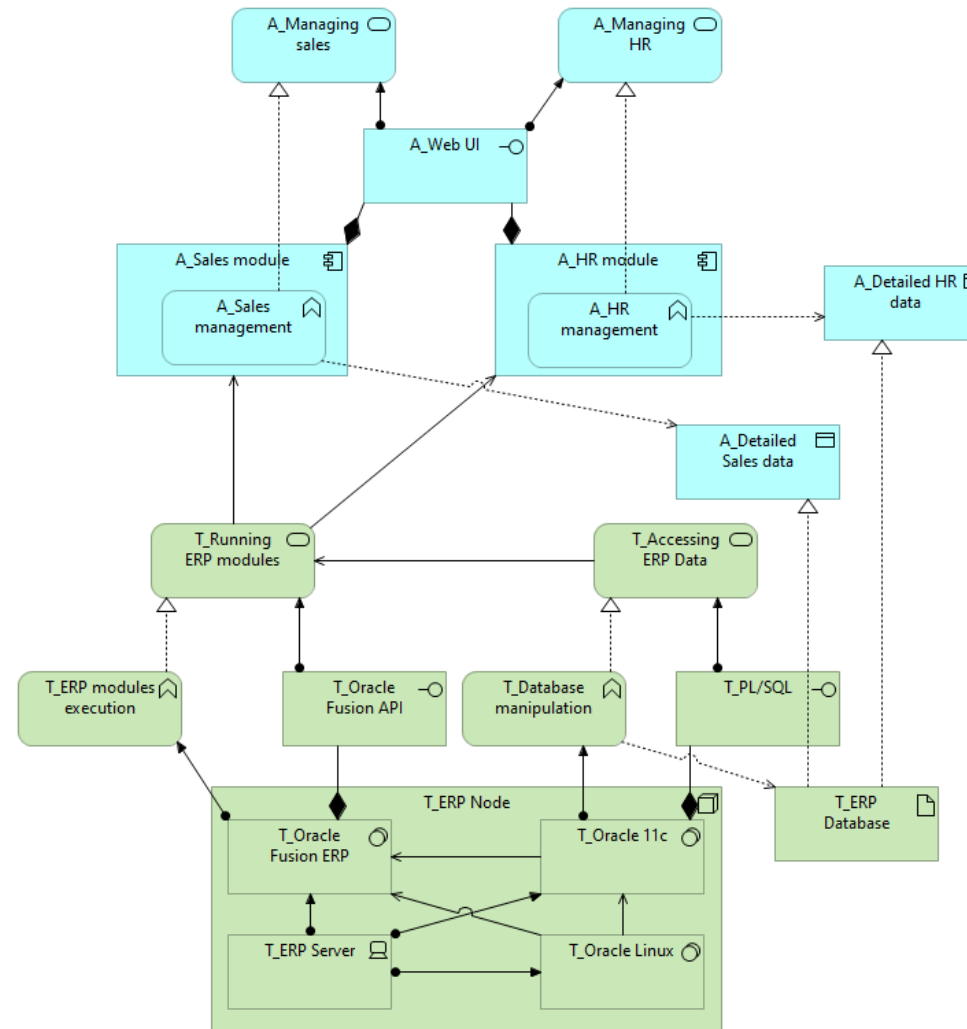
ArchiMate models representing the business goals and the existing ERP system are enclosed in the next slides.

Starting from these models, create a complete ArchiMate model that: 1) fully represents the business layer and links it to the application layer, 2) extends the application and technology layers by adding the elements required for the new reporting service.

# Exercise 1 – Motivation



# Exercise 1 – Existing ERP system



## Exercise 2 – IC

**IC** is an insurance company which wants to offer a new insurance service for small objects (<2000\$) managed completely online for reliable customers.

To achieve this, a customer who wants his assets to be insured has to provide its credentials and photo of the asset and its details (serial number, purchase date) to IC. To ensure that the customer is reliable and the asset inexpensive, IC will then check the customers credentials and past history and estimate the asset's price. If the checks succeed, a proposal will be generated. Otherwise, the request will be rejected.

To support the new service, IC needs to develop a new application with the following functionalities: proposal generation, price estimation, customer reliability check.

The new application will be implemented as a Java EE component running on Apache Tomcat. It will also rely on an operational database deployed on Oracle MySQL. Both the database and the application will reside on a dedicated server, which is connected to the corporate LAN.

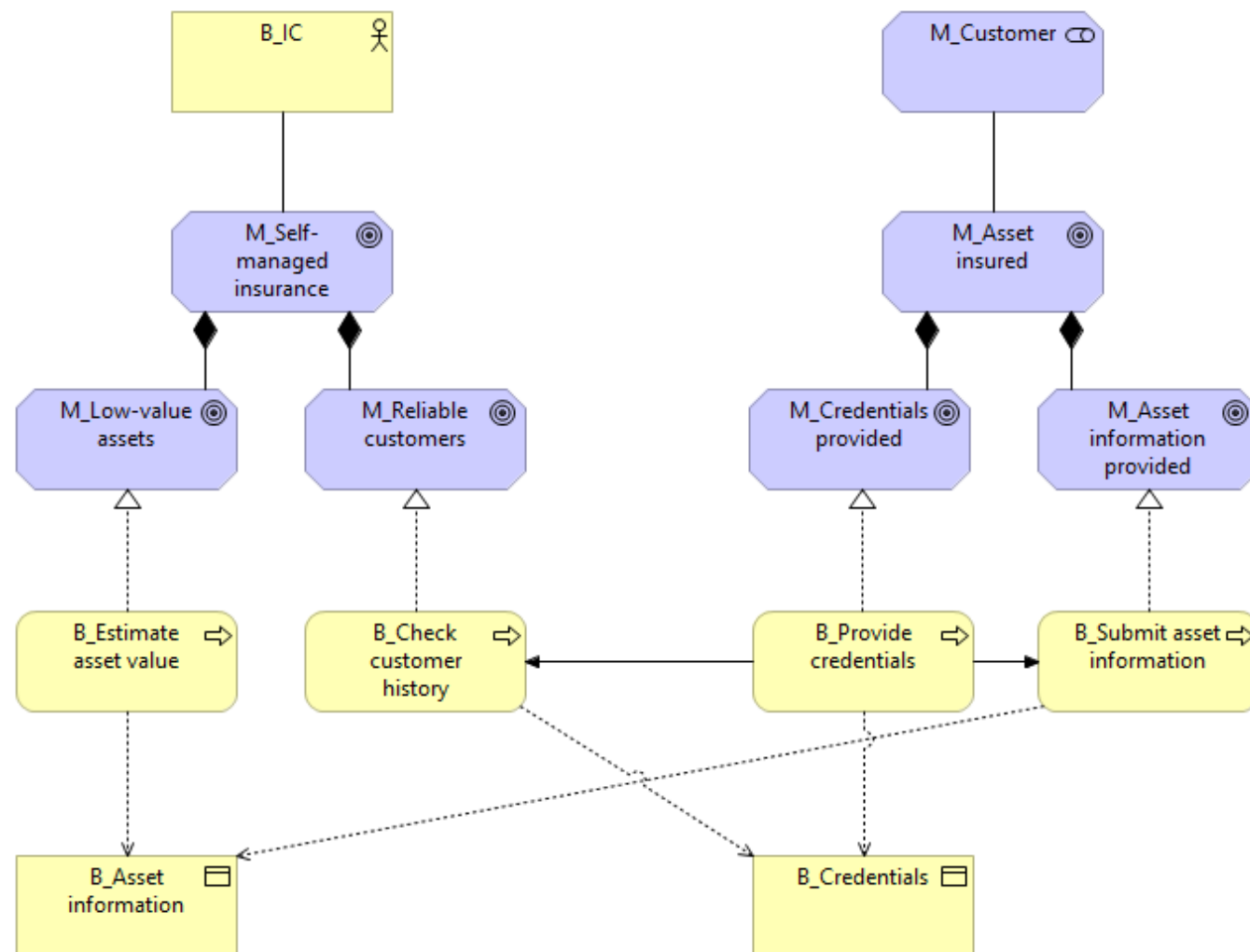
To estimate an item price, an external service will be used.

To check the customer reliability, the application needs data coming from the company's CRM, exposed by the CRM as a REST service and accessed through the corporate LAN.

The corporate LAN is separated from the public Internet by a firewall.

Model in ArchiMate the service provided by the company and its infrastructure.

## Exercise 2 – Motivation





## Exercise 3 – TEL

TEL is a telephony company interested in improving its customer experience. Currently, billing details are only accessible internally by TEL staff.

Data about telephony usage is stored on a Linux server running IBM DB2 UDB database, and it can only be accessed through SQL queries. Conversely, billing information is stored and handled by a legacy transactional CICS application running on an IBM mainframe.

To improve its customer experience, TEL wants to develop a new web portal that can provide real-time billing details to users. In particular, the new portal will provide two main functionalities: inspect billing information and inspect usage information. The process enabled by the new portal will be organized as follows: the user logs in the portal, then he can select an item to inspect from the menu, and finally he can view the billing details in a dedicated page.

The new portal will be built for Microsoft SharePoint Online PaaS cloud service.

To access data from the existing infrastructure, a new node running Microsoft BizTalk Server 2020 middleware will also be introduced. BizTalk Server will offer a gateway service, making CICS applications and relational databases accessible through a standard REST interface.

An ArchiMate model of the existing system is enclosed in the next slide. Extend the model by adding the elements required for the new service.

# Exercise 3 – Existing infrastructure

