# Introducing ArchiMate®

# Organizations and IT

An information system can be defined as “the set of tools, organizational knowledge, and technical skills to manage the information resource”

The computer / IT system is a component of the information system which elaborates, manages and archives information

Business activities can be automated, either fully or partially, or carried out manually (e.g., paper archives, informal documents, decision-making processes)

# Organizations and IT

Information systems are composed of complex applications integrated with each other, aimed at supporting both internal business processes (intra-company) and processes involving collaboration with other companies (inter-company)

Moreover, a company’s information system is often the result of incremental growth and constant evolution (static / dynamic aspect)

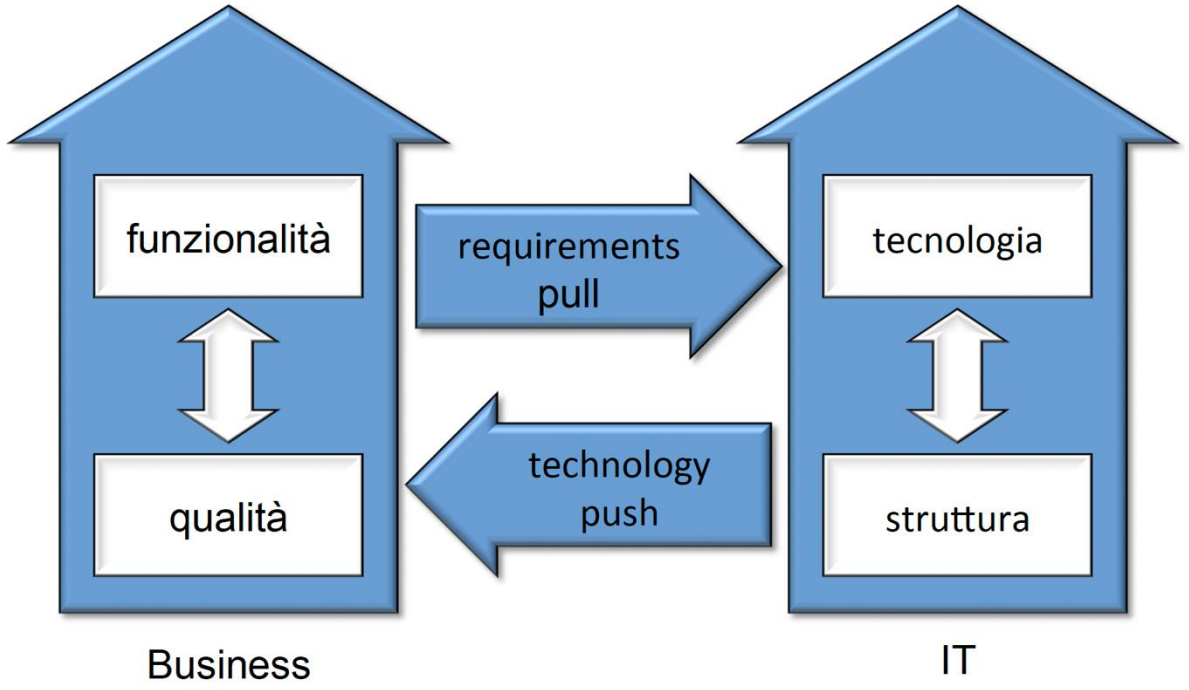
# Organizations and IT


EA is an abstract description of an information system, that can be used to understand its internal structure, support implementation, change and maintenance

It provides an integrated perspective to manage different layers: strategy, business, technology

An architecture gives “viewpoints” to shareholders (owners, users, clients)

# Organizations and IT




Enterprise Architecture modeling is a key factor in maintaining coherence (as-is and to-be) among the components of an enterprise information system, considering several aspects: data, processes, and interactions.

Bi-directional relation among organization and IT: (1) strategic or control decisions impact technological choices, and (2) new technological opportunities motivate changes in objectives, organization and activities

# What ArchiMate is

- ArchiMate is a graphical language for enterprise architecture modeling
- Mainly used to describe building blocks though diagrams
- Maintained by TheOpenGroup Consortium (the same as TOGAF)
- Now in version 3.2 (3.1 in course)
- The specification associates symbols with their semantic
- https://pubs.opengroup.org/architecture/archimate3-doc/

< Previous ▲ Home Next >

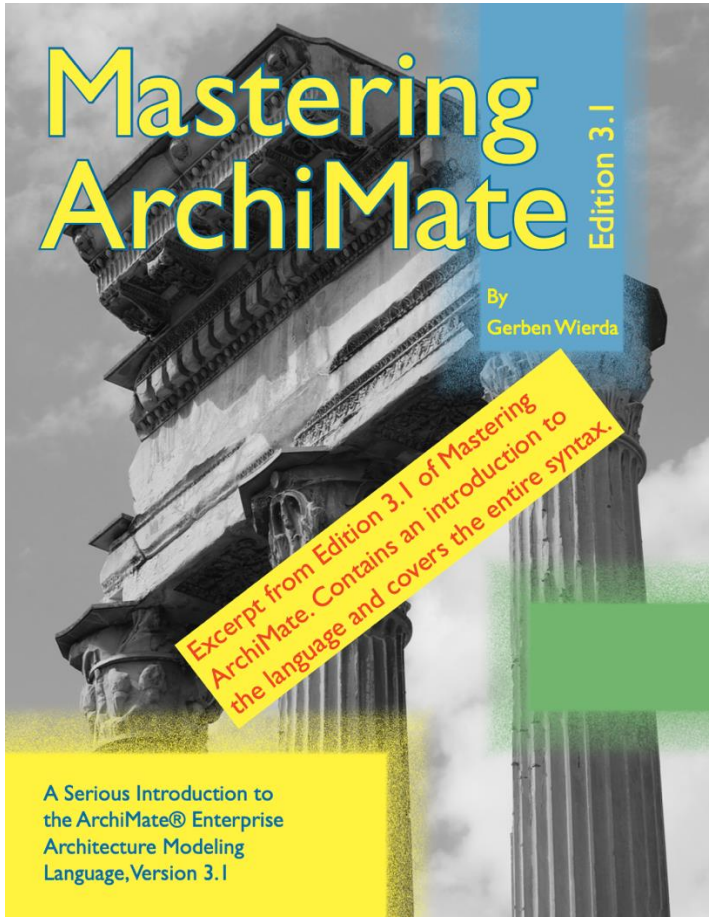
ArchiMate® 3.1 Specification
Copyright © 2012-2019 The Open Group
Previous versions: [3.0.1 | 3.0 | 2.1 ]

THE Open GROUP

DOWNLOAD

Welcome to the ArchiMate® 3.1 Specification, *a Standard of The Open Group*

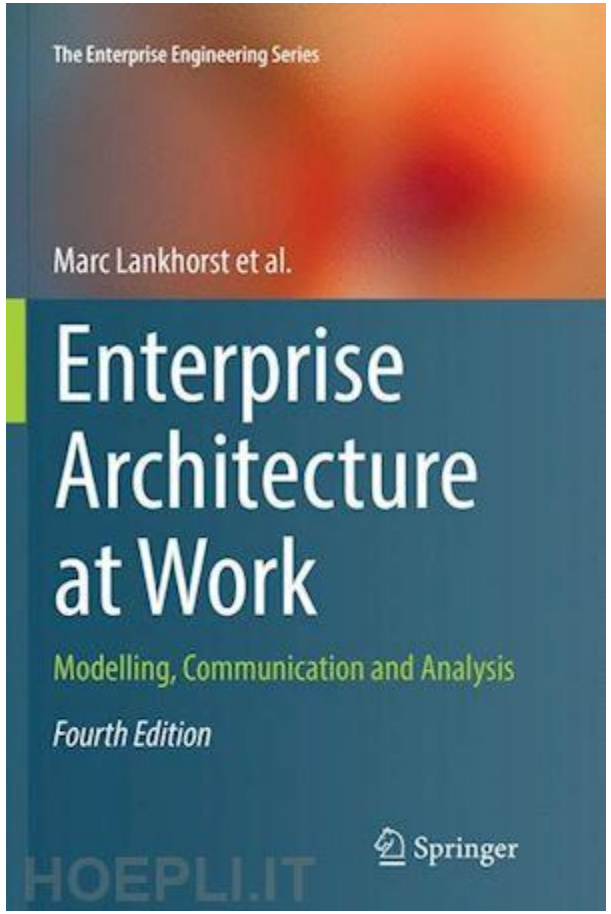
# Book suggestion 1




Free introductory chapter freely available at: https://ea.rna.nl/the-book-edition-iii/

Please take care of the colours used: Archimate is “colorless”, the use of yellow, green and blue is arbitrary

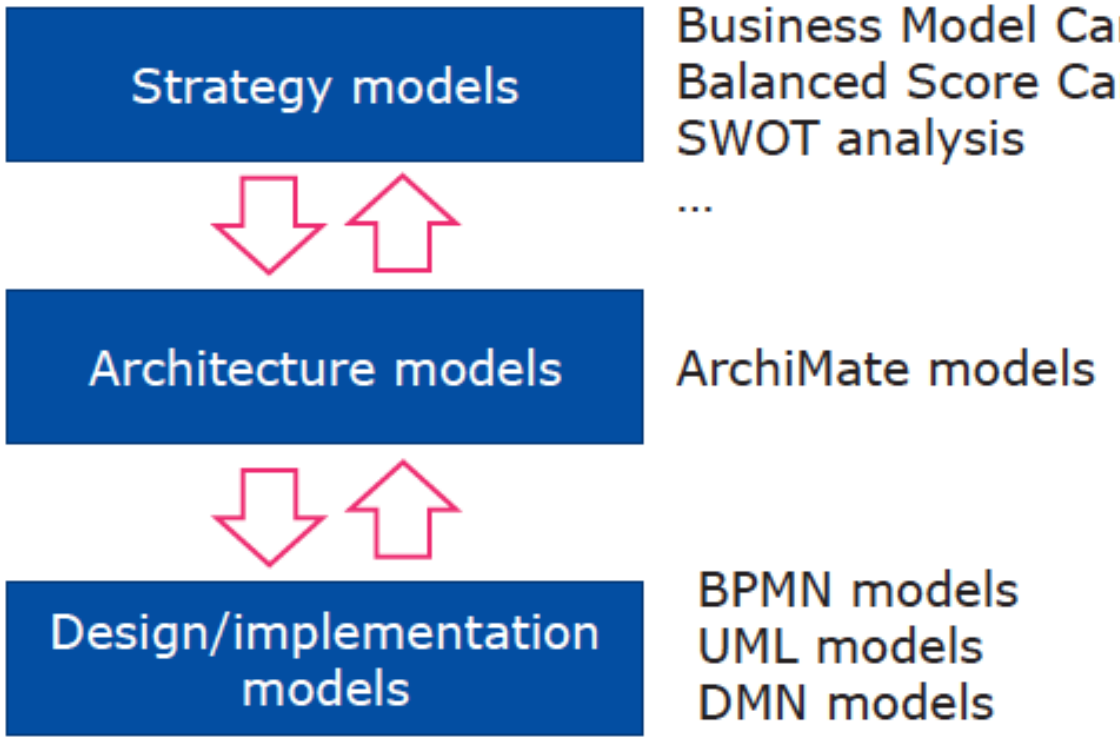
# Book suggestion 2




Covers all the aspects about Enterprise Architectures using Archimate as modeling language

(free version of a previous edition through Polimi library service)

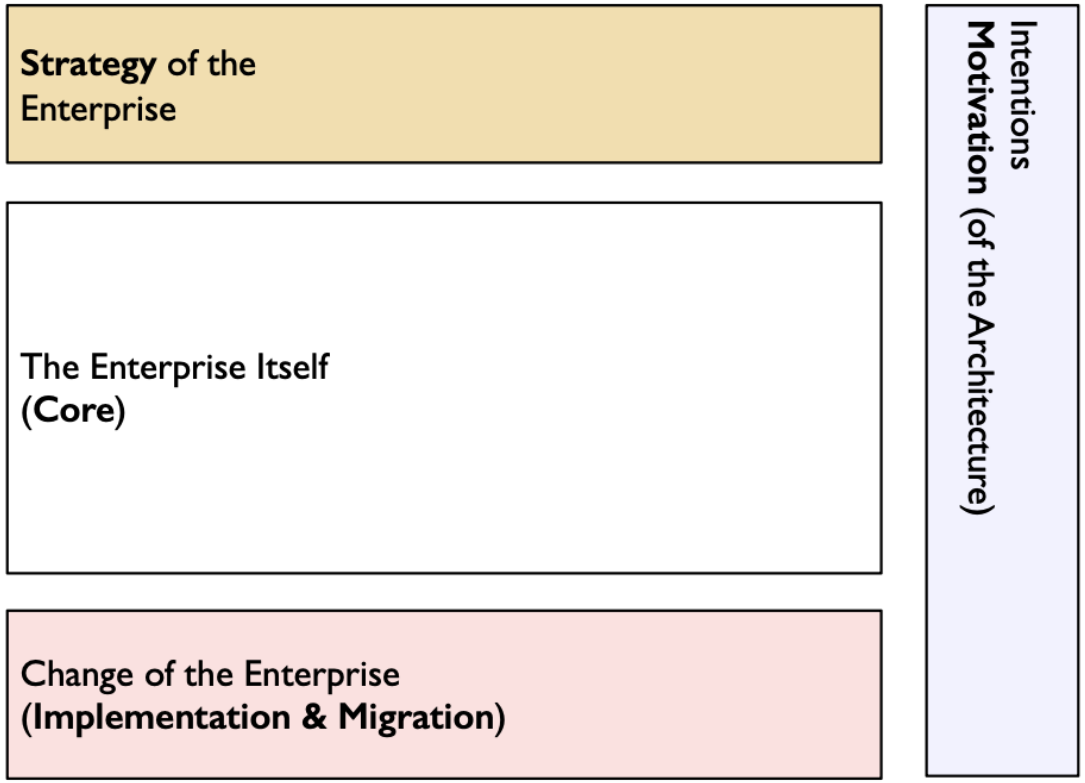
# Positioning ArchiMate




Mediation among business and IT world that are historically “separated”

Main viewpoint for us: technologic, capturing the business needs

# ArchiMate map

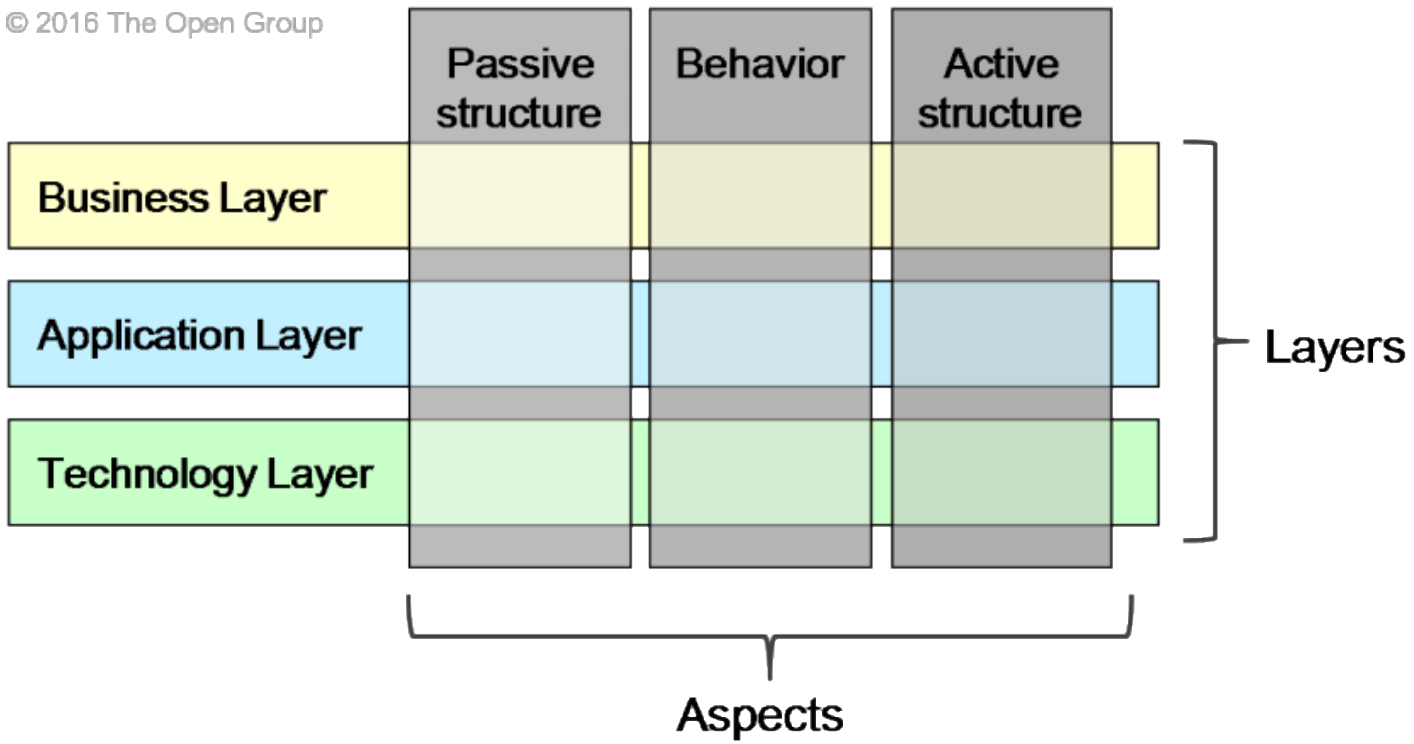



View 1. *The ArchiMate Map — Rough*

The EA description is provided through a series of diagrams, consistent with each other

Each diagram is focused on describing the system according to the interests of specific stakeholders (strategic, goals for as-is, change)

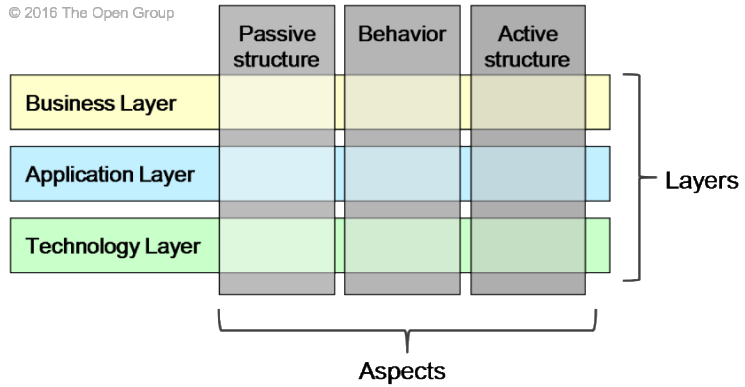
# ArchiMate CORE framework




ArchiMate Core (main course focus) describes different layers/domains of the EA:

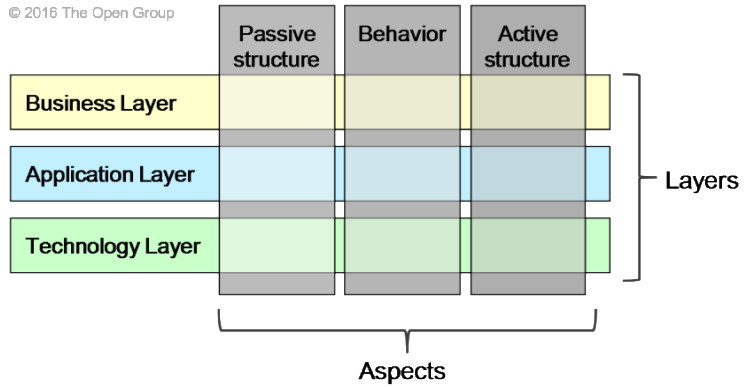
- Business: linked to objectives and requirements
- Applicative: identifying functionalities of systems
- Infrastructure: documenting technological choices (eg data interchange platforms, security...)

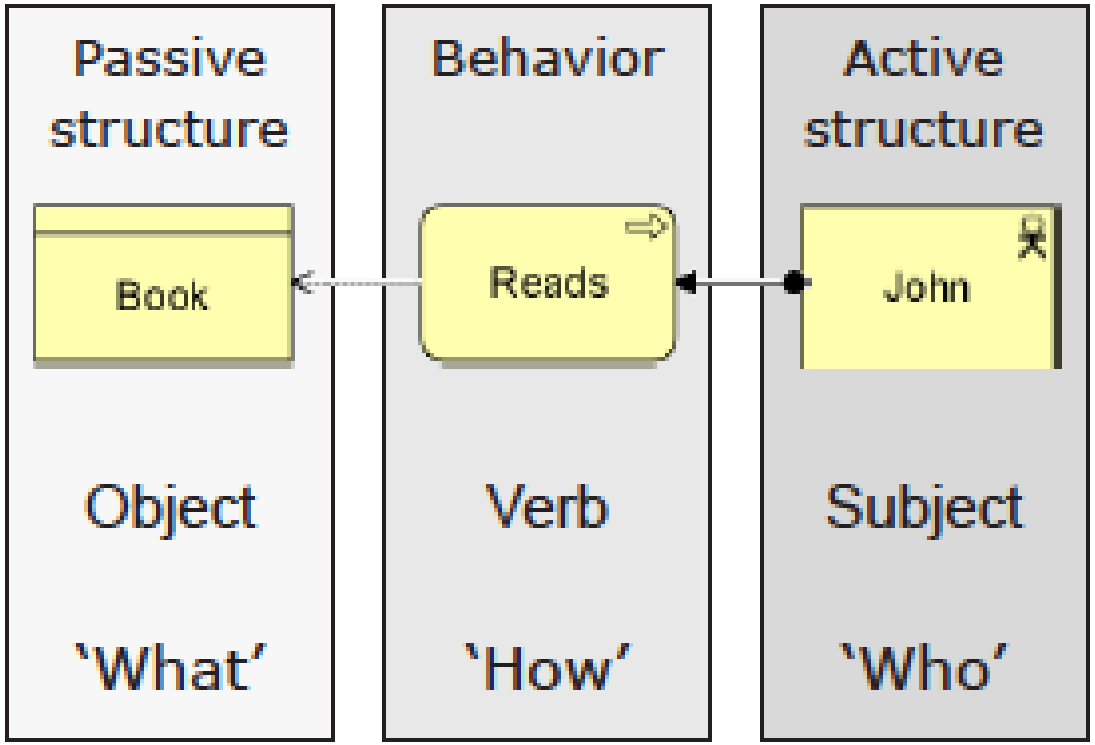
# Layers




- Business layer:
  - Offer products and services to external consumers
  - A business stakeholder needs to know the nature of the service to be offered, to whom it is offered, how it is delivered, and which business functions are involved
- Application layer:
  - Support the business layer with application services – realized by software applications
  - An application stakeholder, a developer or system integrator, needs to understand, which applications and data are involved, need to be developed and stored
- Technology layer:
  - Offers infrastructure services needed to run applications, realized by computer, clusters, and communication
  - An infrastructure stakeholder needs to have a clear view of the platforms to be used to install and operate the applications that provide the business services.

# Aspects



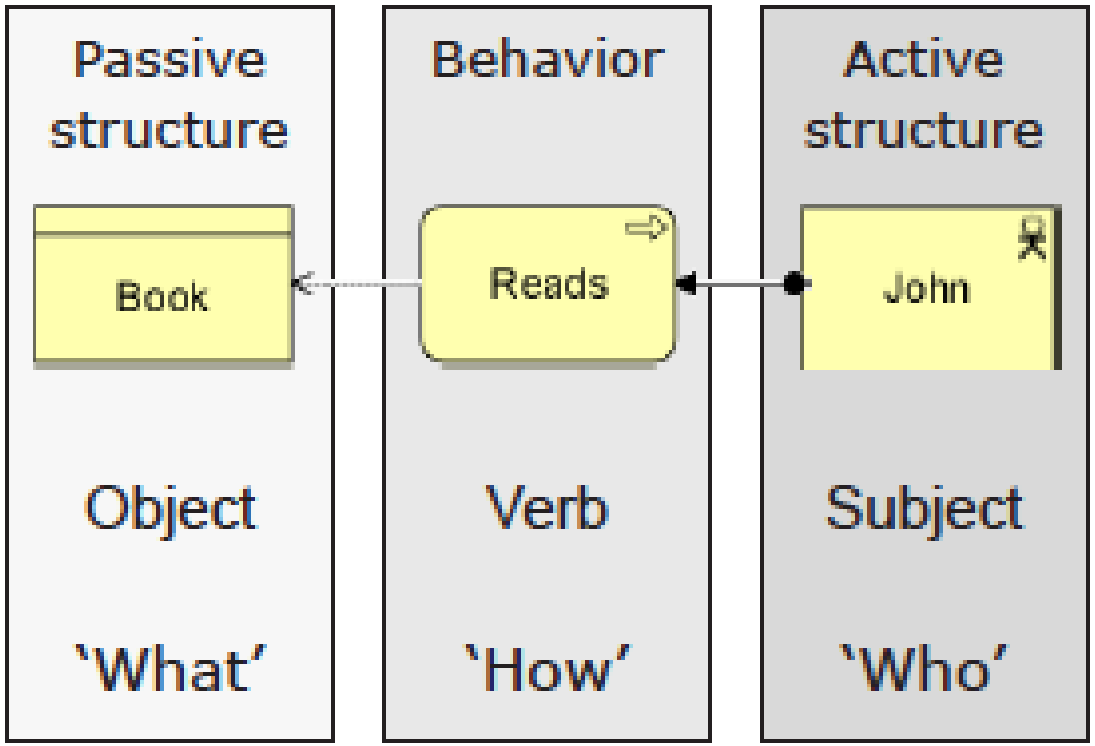

Each layer can be described through three aspects:

Active elements ("subject"): who provides the behavior, like actors or devices

Behavioral elements ("verb"): Behavior describes what a system does

Passive elements ("object"): The object of the behavior

# Aspects




Basic example in yellow:
"*John reads a book*"

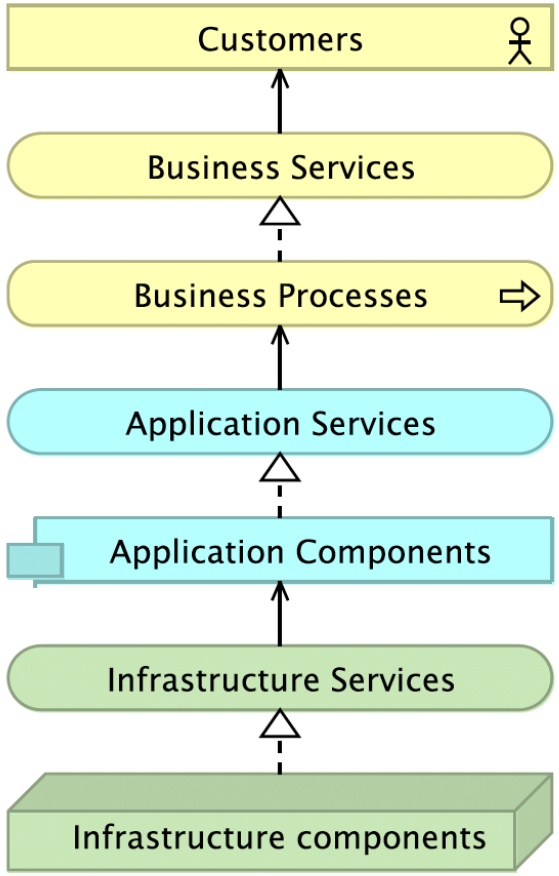
The container symbols correspond to aspects (grammar). ArchiMate also describes a syntax and a semantic (eg. the meaning of arrows)

An active element can be linked only to a behavioral or another active element

A behavioral element can be linked to any element

A passive element can be linked only to a behavioral element

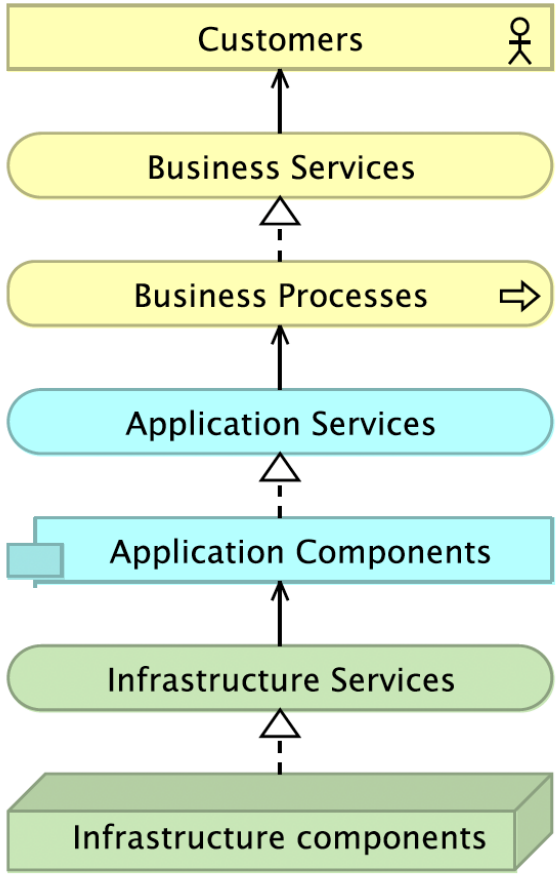
# Service orientation and layering




Each layer (domain) of the ArchiMate Core framework has a service-oriented structure, with an emphasis on the separation between interface and implementation ("OOP-like")

- The service interface: above the "visibility line", exposed to the next level who consumes the service (serving relation)
- Service implementation: how the service is implemented (realization relation)

# Service orientation and layering




The business layer is the one that defines the service as it is seen by the end user. The layered architecture ensures that the final consumer only sees the business service and has no awareness of what happens in the underlying layers

The end user interacts with the interface of business services, which are implemented by business processes

In turn, business processes are the consumers of application services, which are implemented by application components

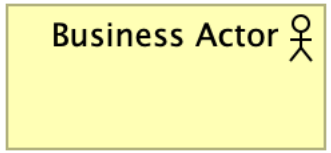
Finally, the application components rely on infrastructure services

# Modeling tools

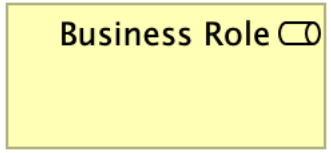
- Archi (next slides are based on it):
  - https://www.archimatetool.com/
- Signavio (academic version is available for free if registered with a University account)
  - https://academic.signavio.com/p/login
- Stencils for Microsoft Visio:
  - https://publications.opengroup.org/i163

# Archimate®
# Business Layer

# Active elements

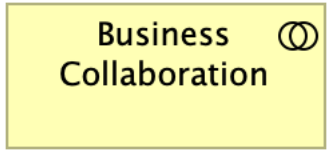



A business entity that is capable of performing behavior (it can be associated to actions)




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

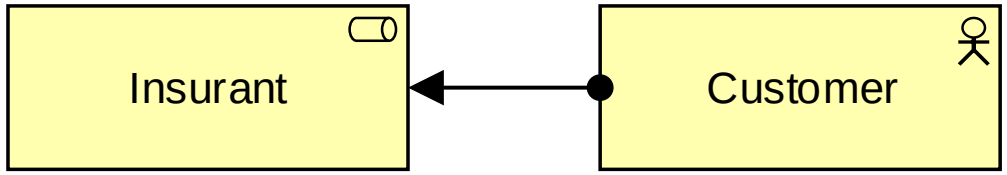

A point of access where a (business) service is made available to the environment / users (examples: website, call center)




An aggregate of two or more business active structure elements, internal to the organization, that work together to perform a collective behavior

# Roles and actors

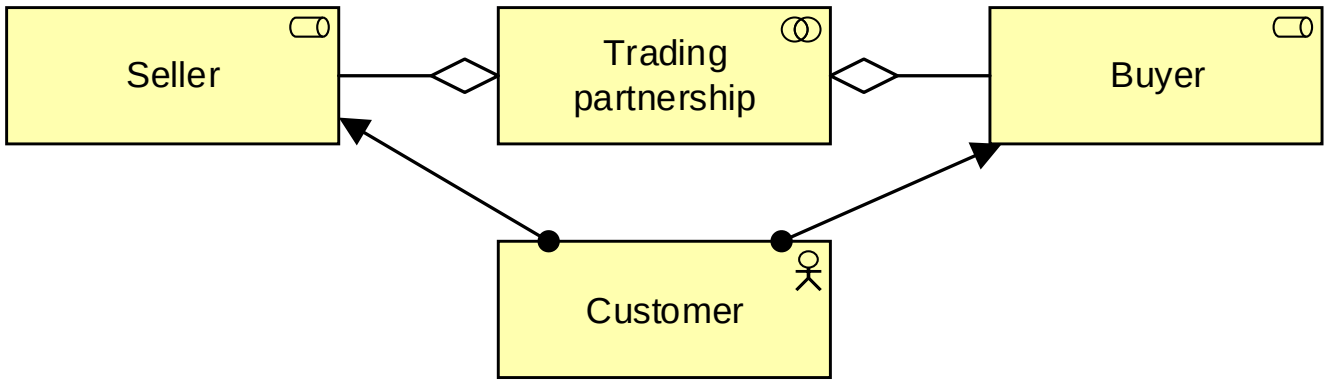
- Roles can be covered by different actors




- “Assigned to” means that a role can be filled by that actor

# Interaction through collaboration

- A collaboration is identified by the “Aggregation” relation




- Empty diamond at the end (full diamond means “Composition” instead)
- Actors survive the resolution of the Aggregation relation (weaker), while Composition is bound to the lifecyle of the related actors

# Interaction through a service

- A Service example: a Seller offers a Service through an interface with a unary composition (no sellers implies no interface)
- “Composed of” relationship indicates that an element consists of one or more other concepts (structural part/whole)


- The Web store Serves a Buyer (arrow towards the served actor)

# Behavioral elements

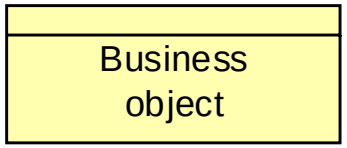

A sequence of business behaviors (activity-like) that achieves a specific outcome


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

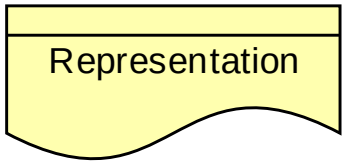

An explicitly defined, externally exposed business behavior


A business behavior element that denotes a state change

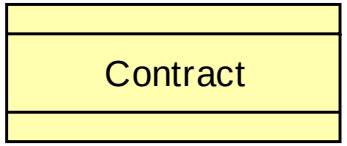
# Passive elements



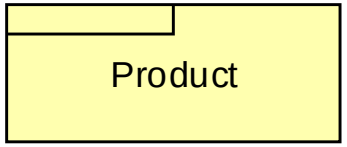

A relevant concept used within a particular business domain




The perceptible form of information carried out by a business object

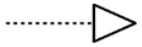
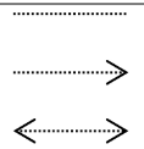
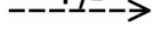
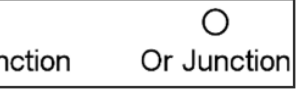



A formal or informal specification of an agreement between a provider and a consumer

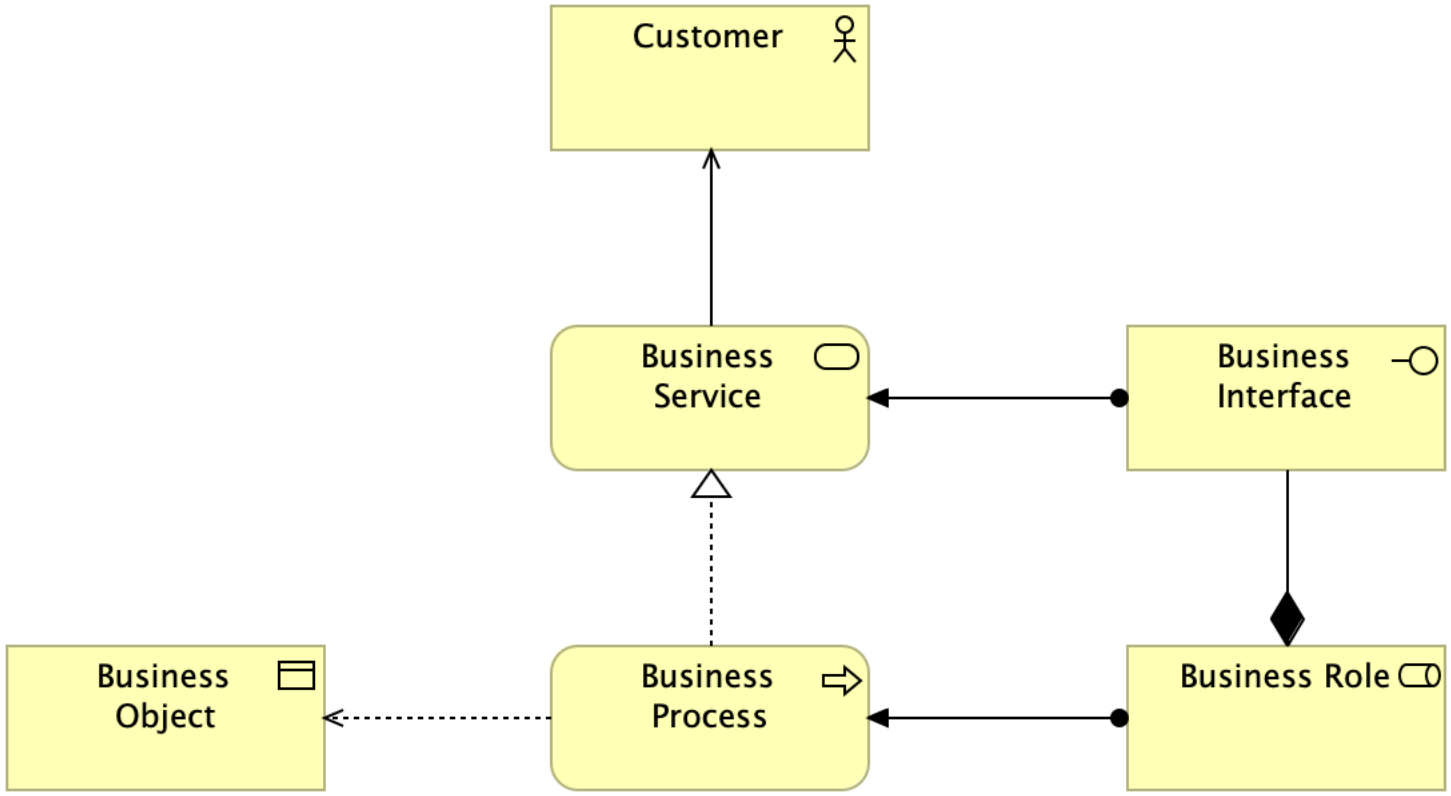



A collection of services and passive structure elements, which is offered to customers

# Relationships

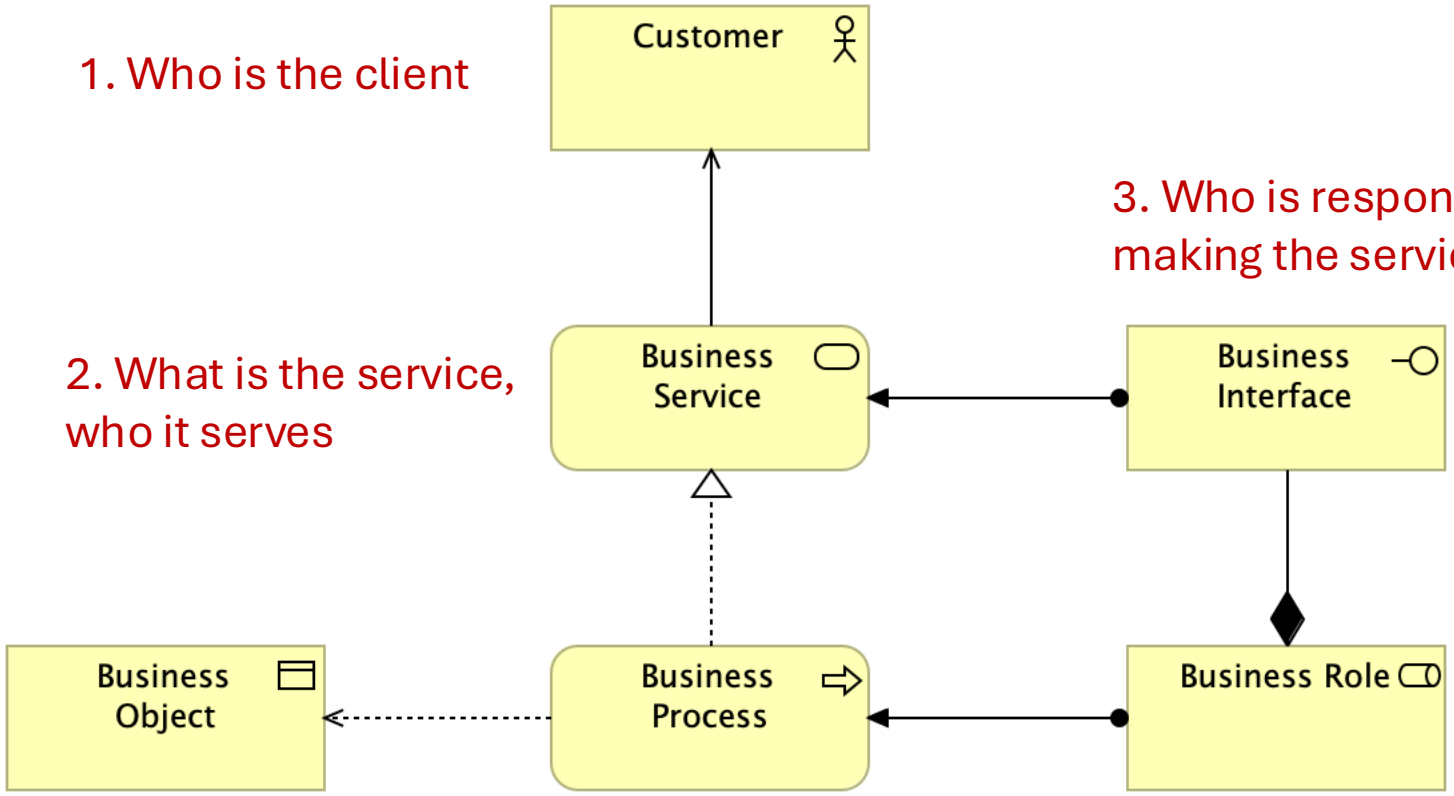
| **Structural Relationships** | | **Notation** |
|---|---|---|
| Composition | Indicates that an element consists of one or more other concepts. |  |
| Aggregation | indicates that an element groups a number of other concepts. |  |
| Assignment | Expresses the allocation of responsibility, performance 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. |  |

# Basic business pattern



# Basic business pattern

1. Who is the client




3. Who is responsible of making the service available (Assignment)

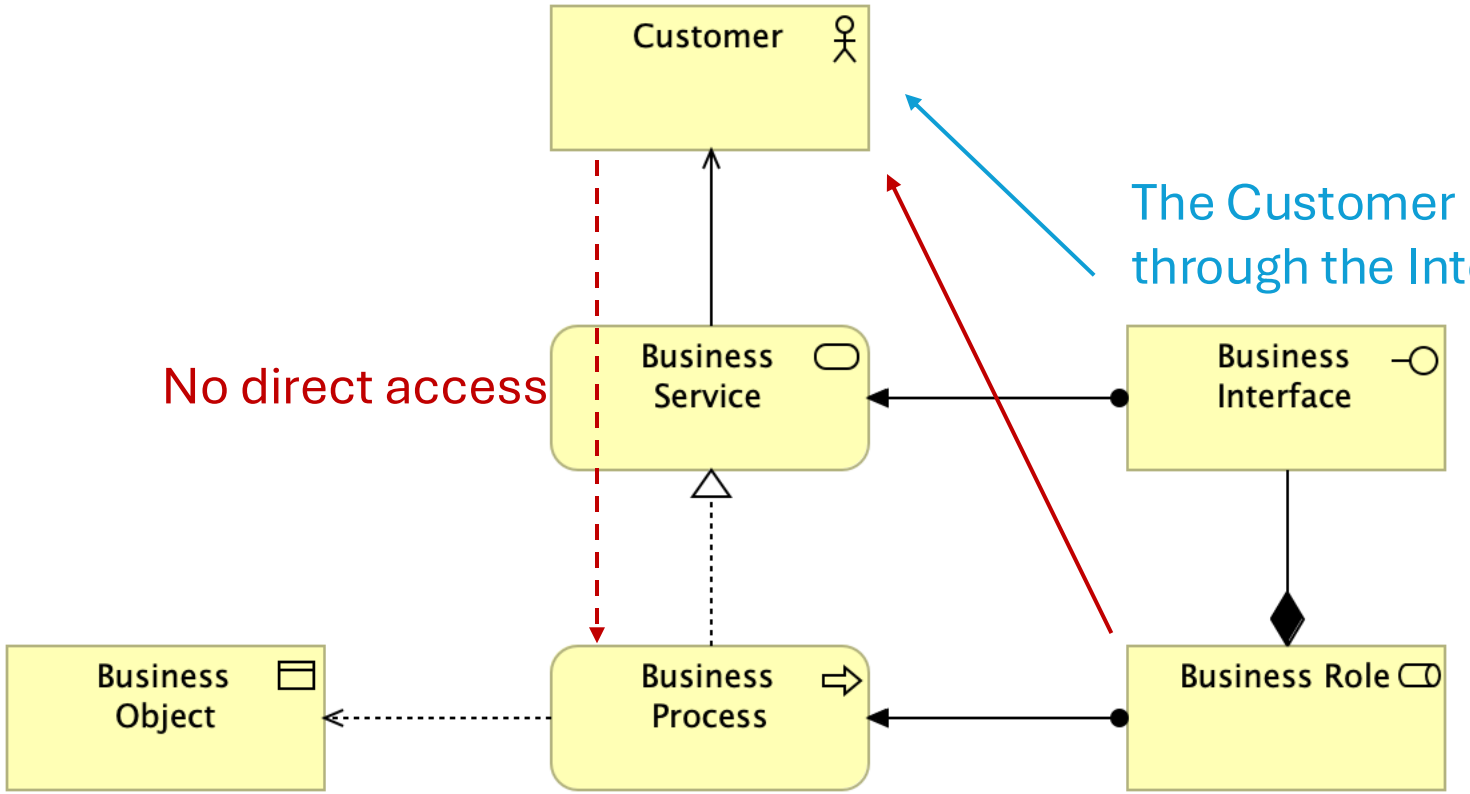
2. What is the service, who it serves

6. The business object accessed

4. Who implements the service (Realization)

5. Who is responsible of running the Process (and is available through the interface)

# Basic business pattern



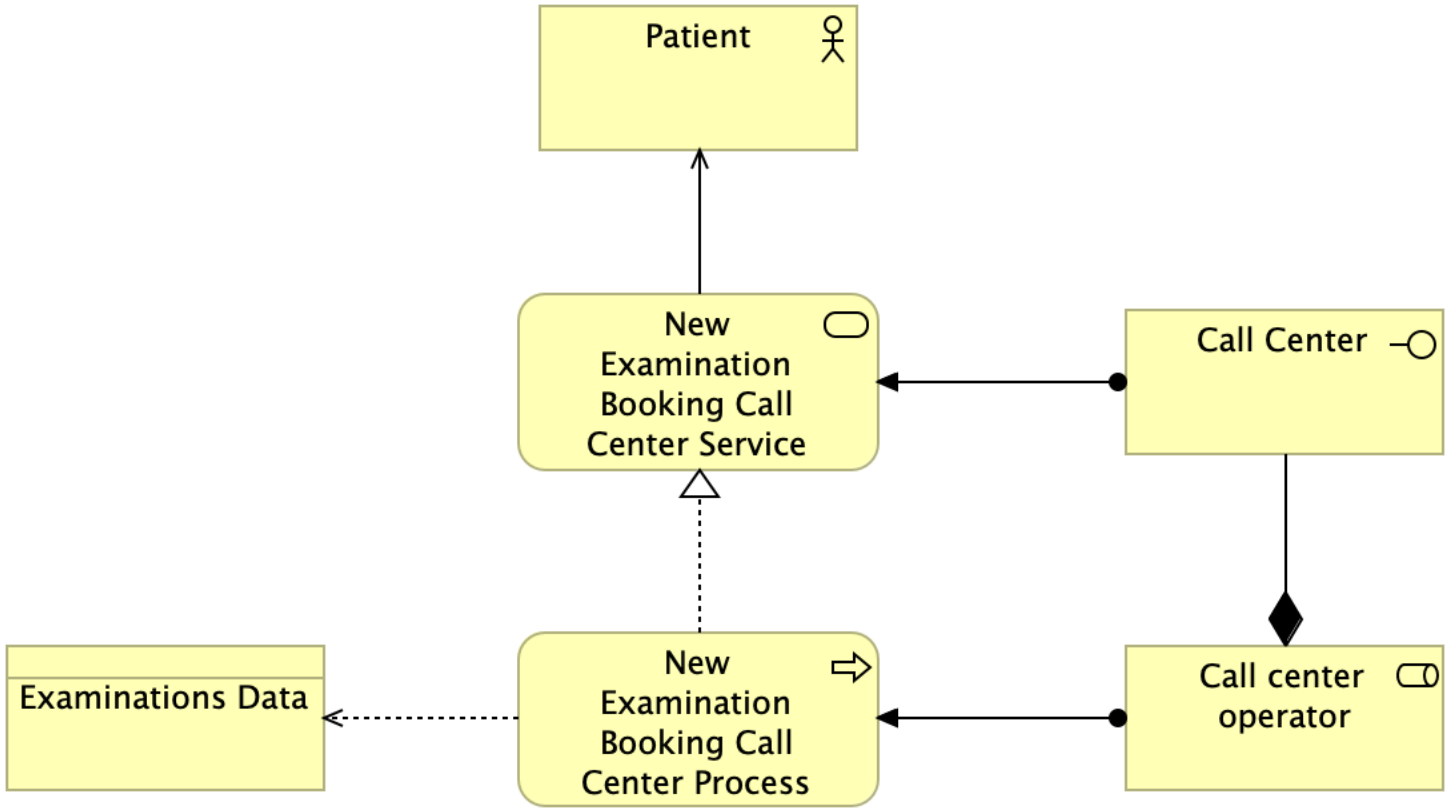

The Customer accesses the service through the Interface

No direct access

How the role interacts with the customer is specified by the interface

# A first example

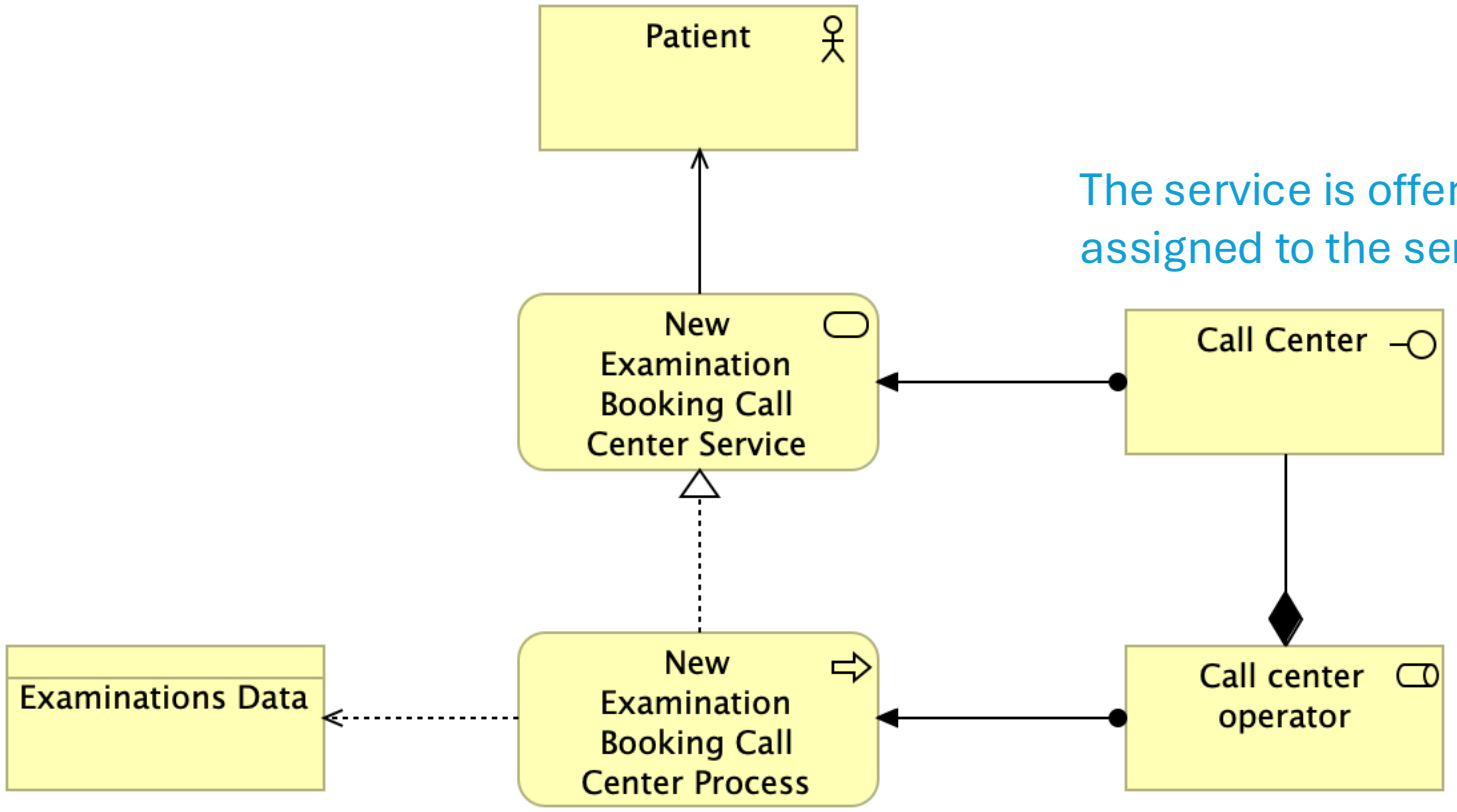
Exam booking service



# A first example

Exam booking service

The service is offered to patients

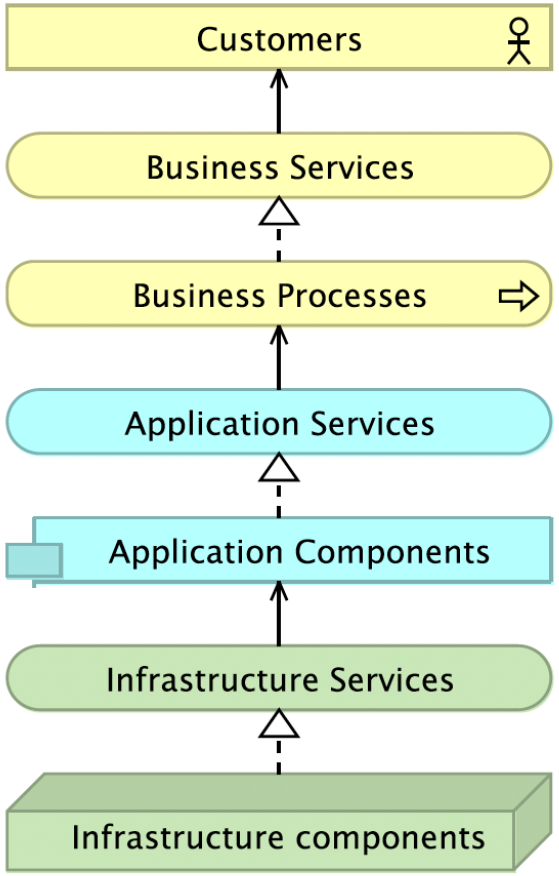


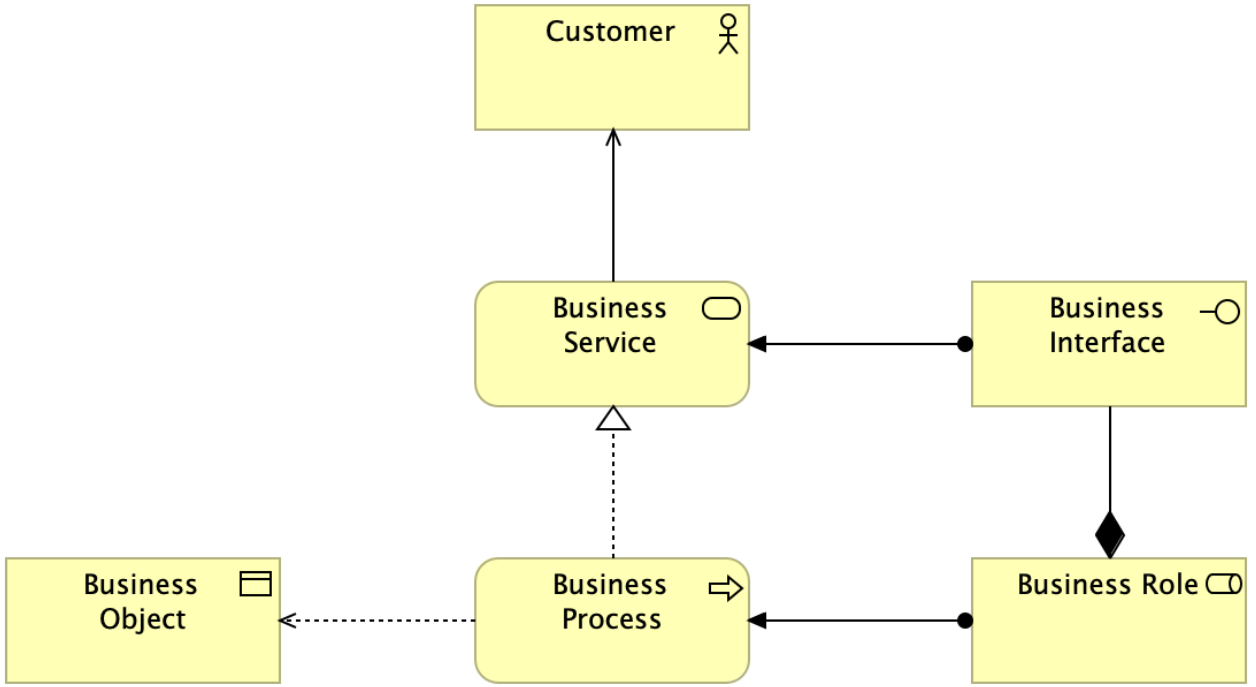

The service is offered via a call center, assigned to the service

A generic process implements the service

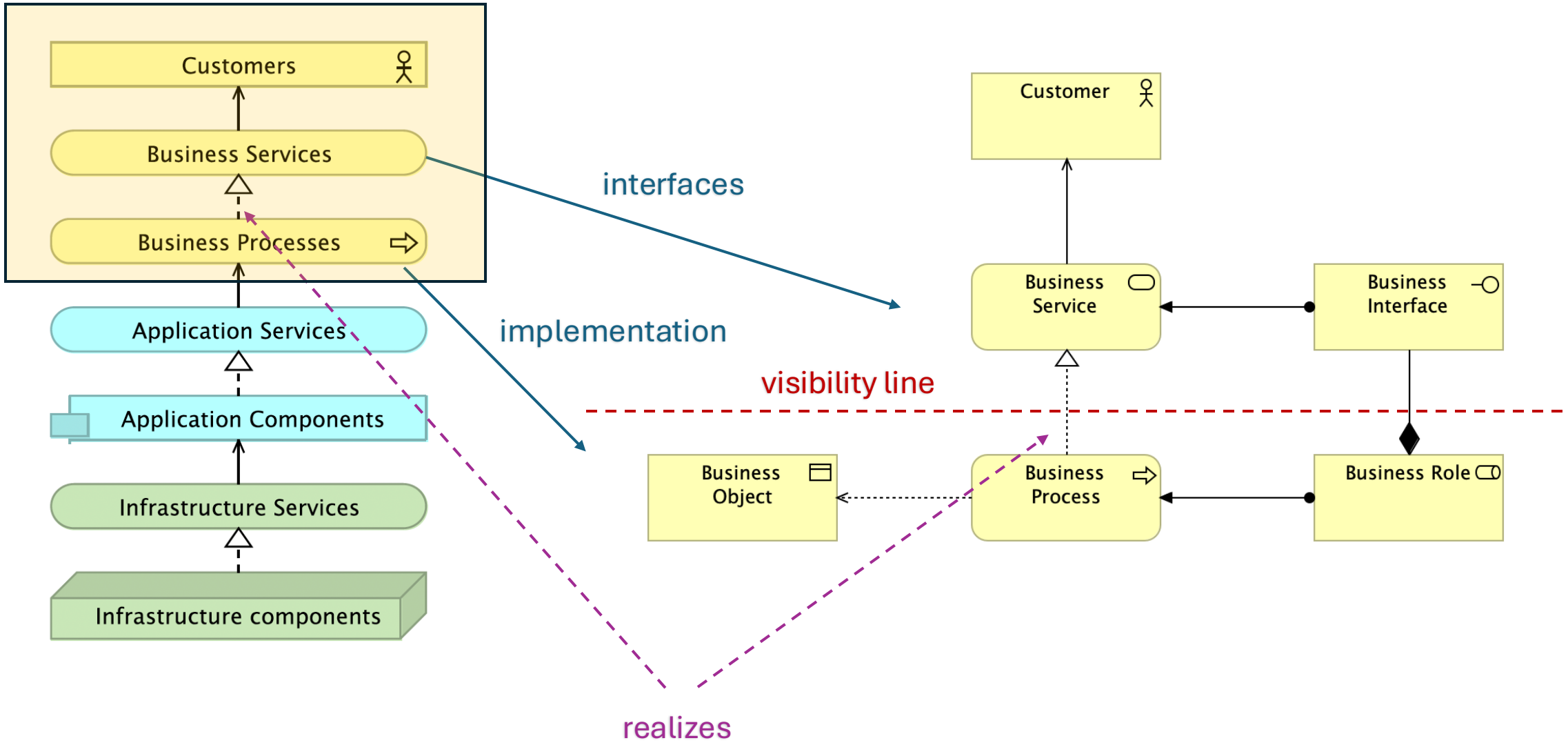
An operator is responsible for executing the process

# Basic business pattern – Layered view



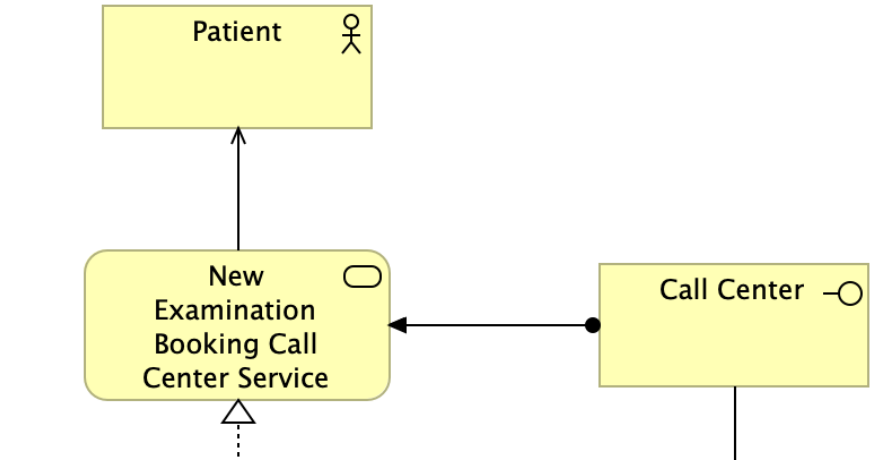



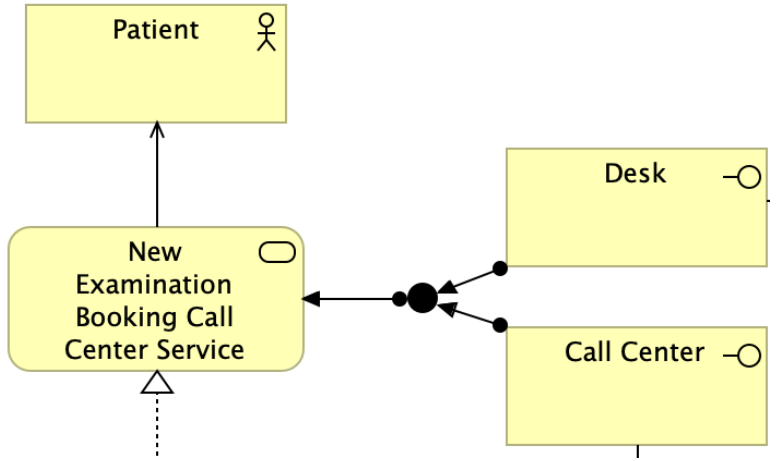
# Basic business pattern – Layered view



# Above the visibility line

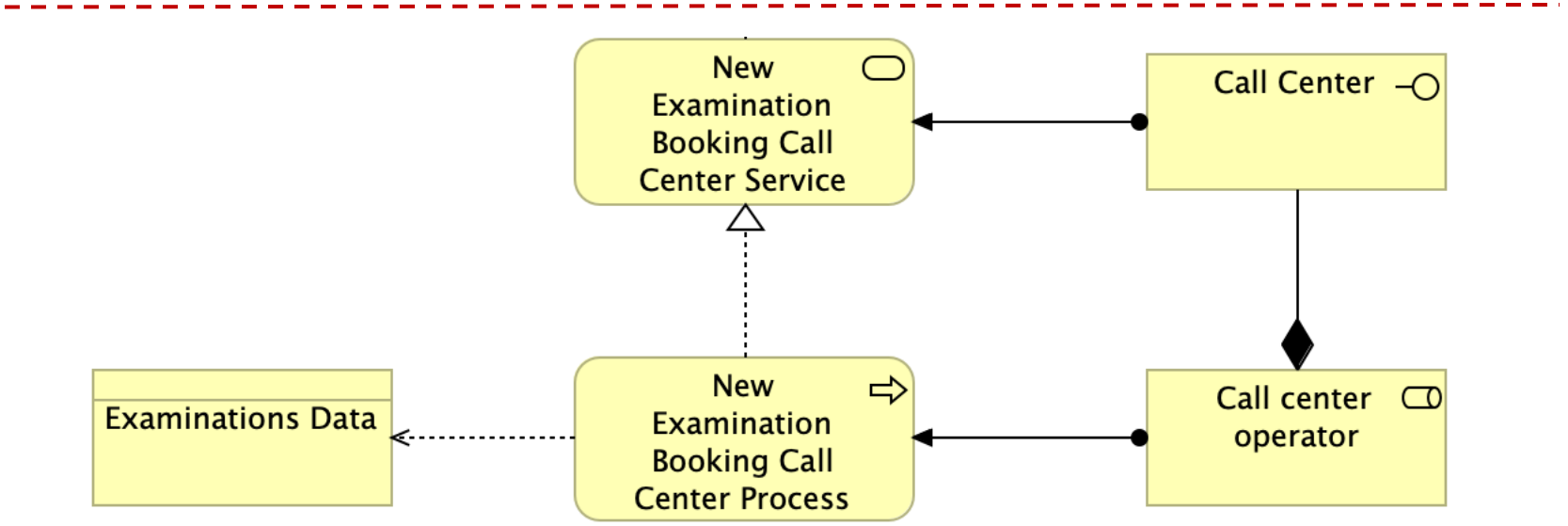
- The user “sees” a service, accessible through an interface
- Interfaces can be more than one: use generic junctions for same-type relations: AND (full circle) or XOR (empty)



# Behind the visibility line

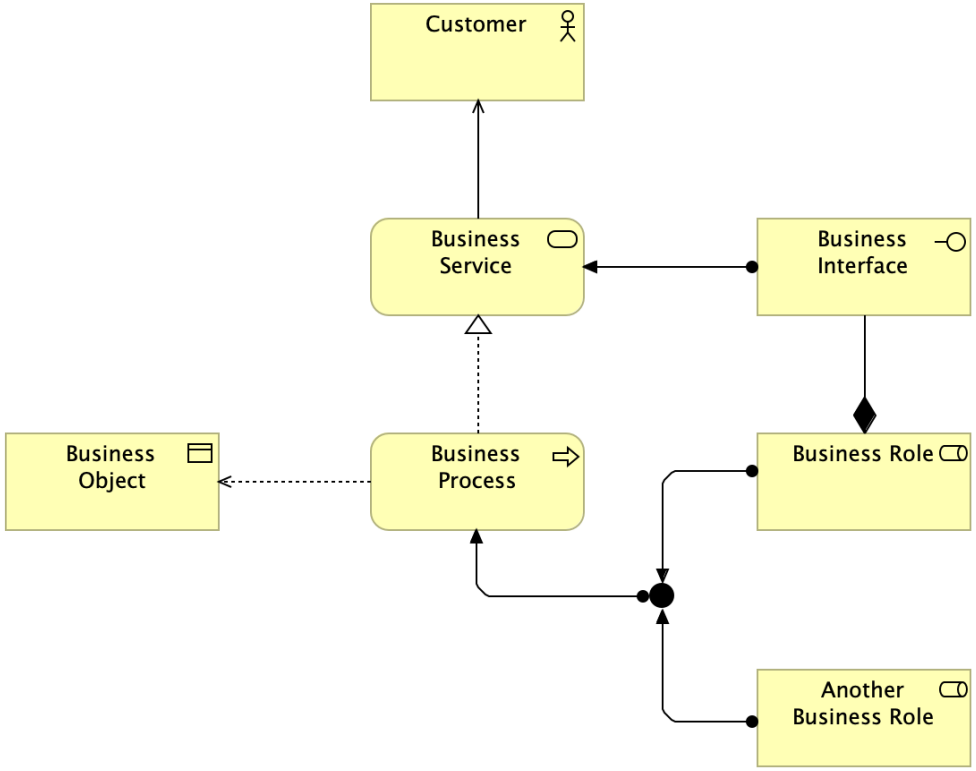
- The process that realizes the service is defined together with the actors/roles involved
- The role assigned to the process knows the process

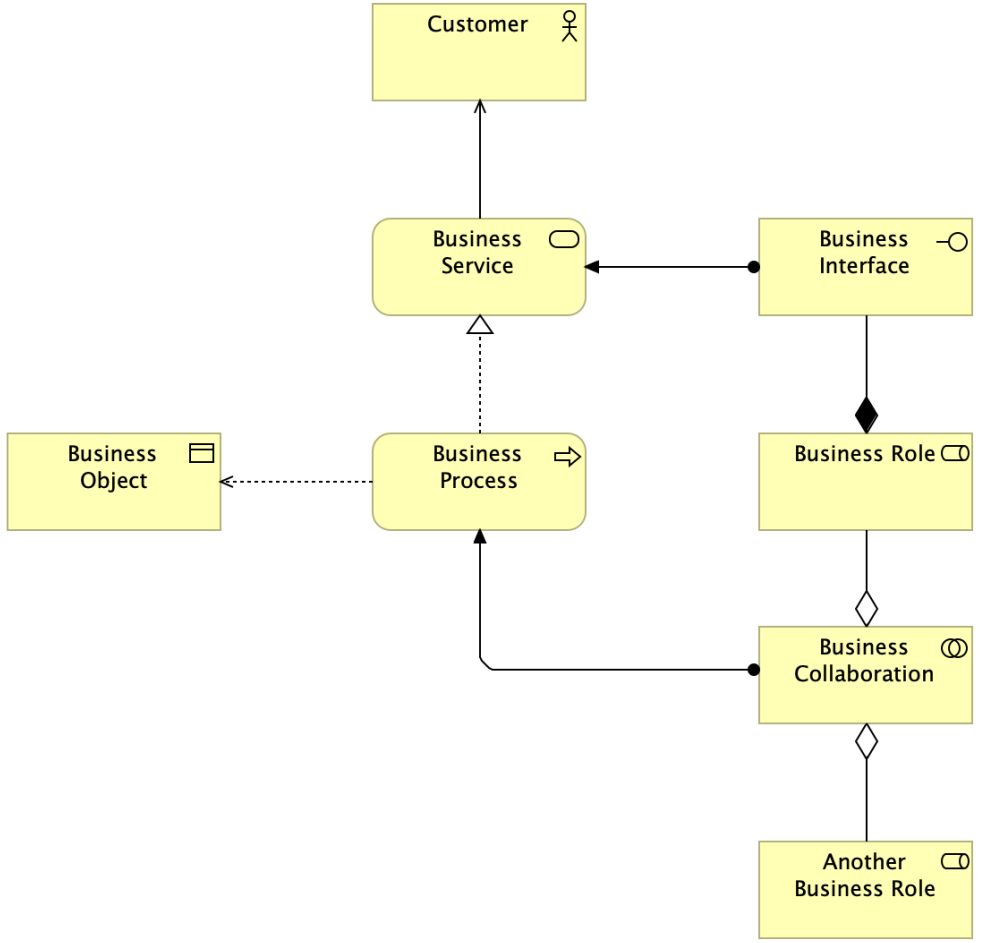


# Role assignment

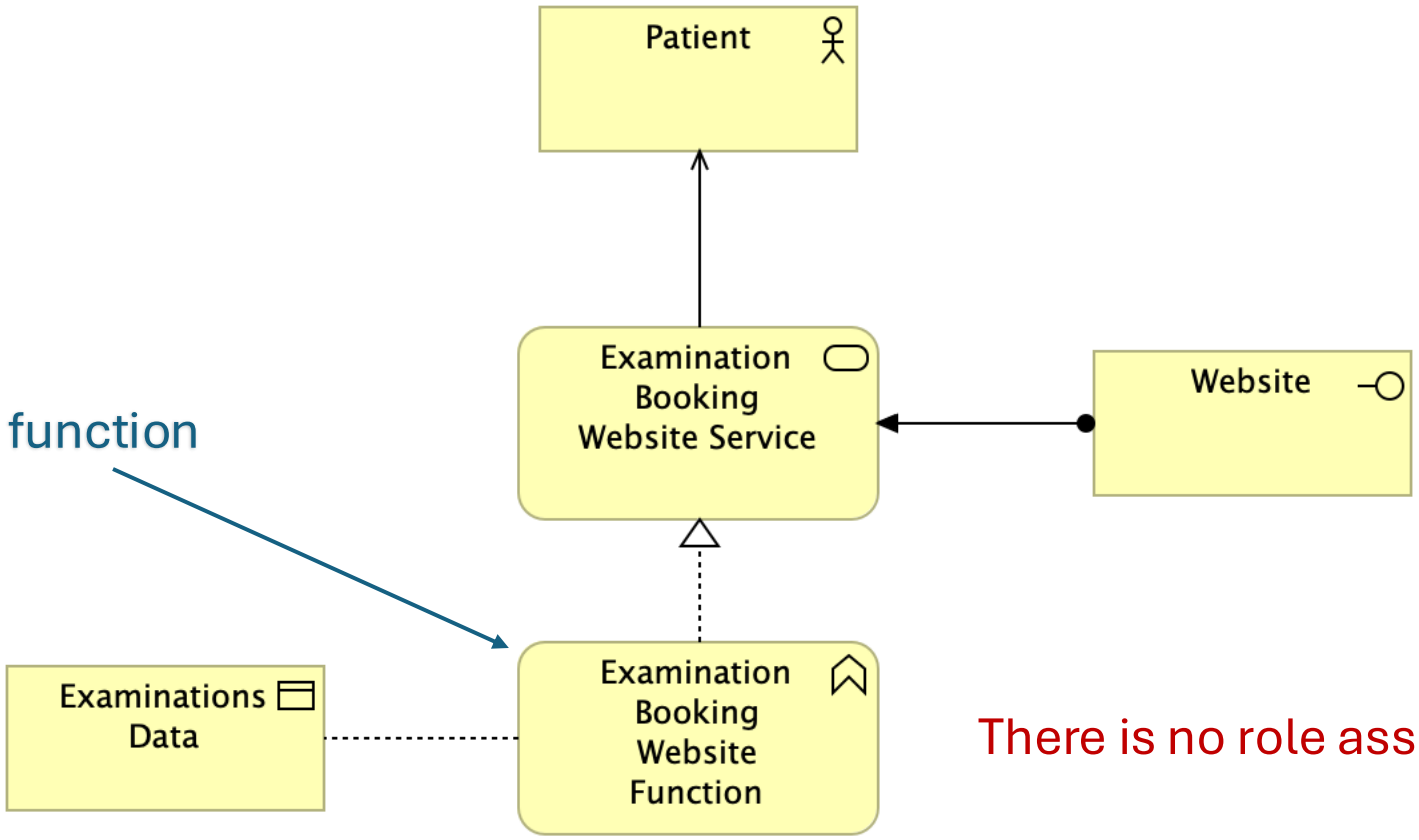
- There must be at most one role assigned to a process
- Junctions or collaborations can be introduced to manage multiple assignments
- In case of no assignments, the process is considered as fully automated

# More than one responsible role



# A self-serve service




There is no role assigned to the process!

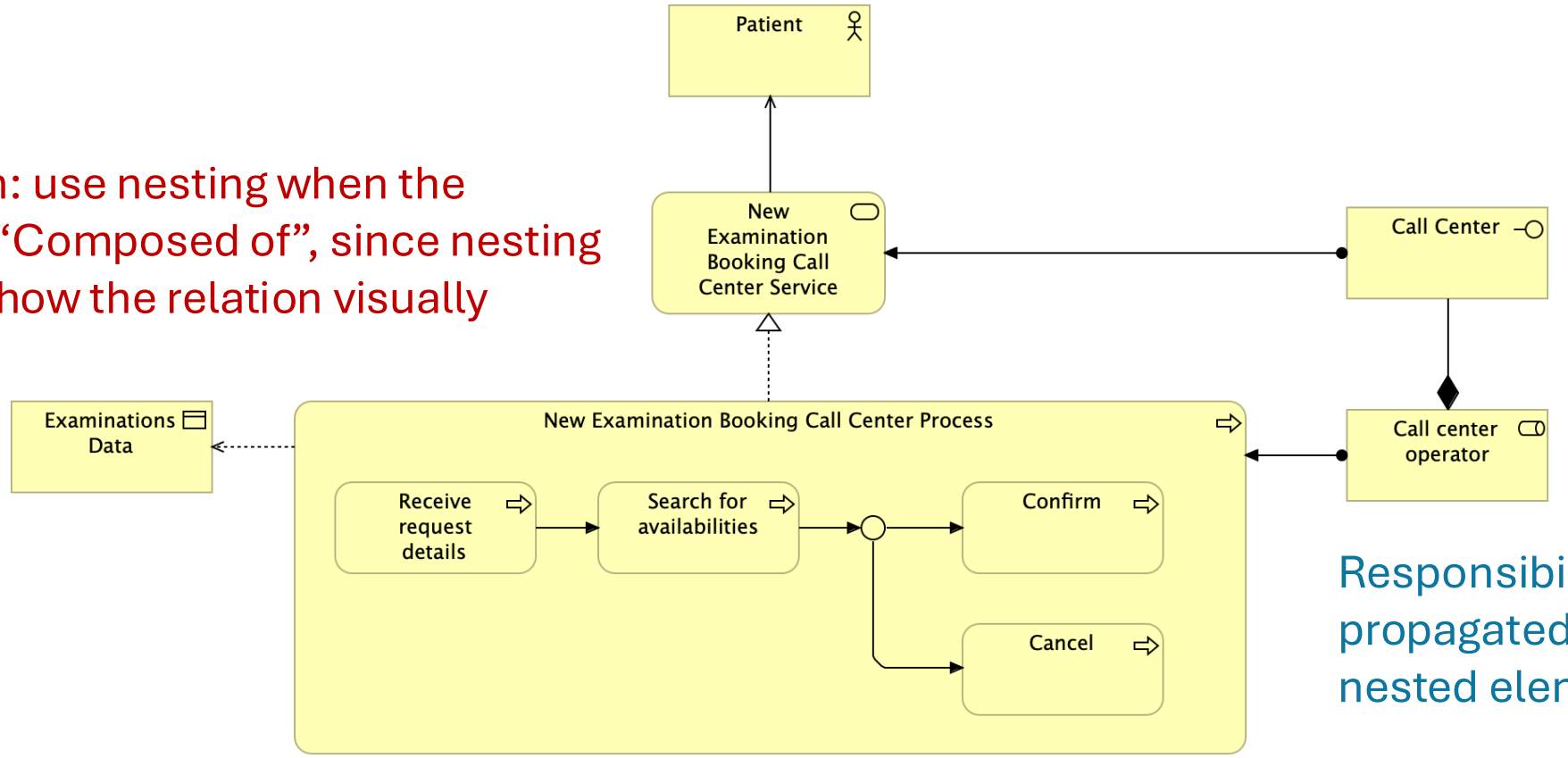
# Functions and processes (business layer)

- Functions and processes are performed by a single role
  - If more roles are involved use either a junction or an interaction (which is performed by a collaboration) to make it more explicit
- Functions and processes can be composed of other behavioural elements, including functions and processes
- Functions are used when:
  - The composing elements have something in common (e.g., the role, the resources) but no specified order
- Processes are used when:
  - The composing elements collectively, executed in order, achieve a goal

# Process specification

With nesting

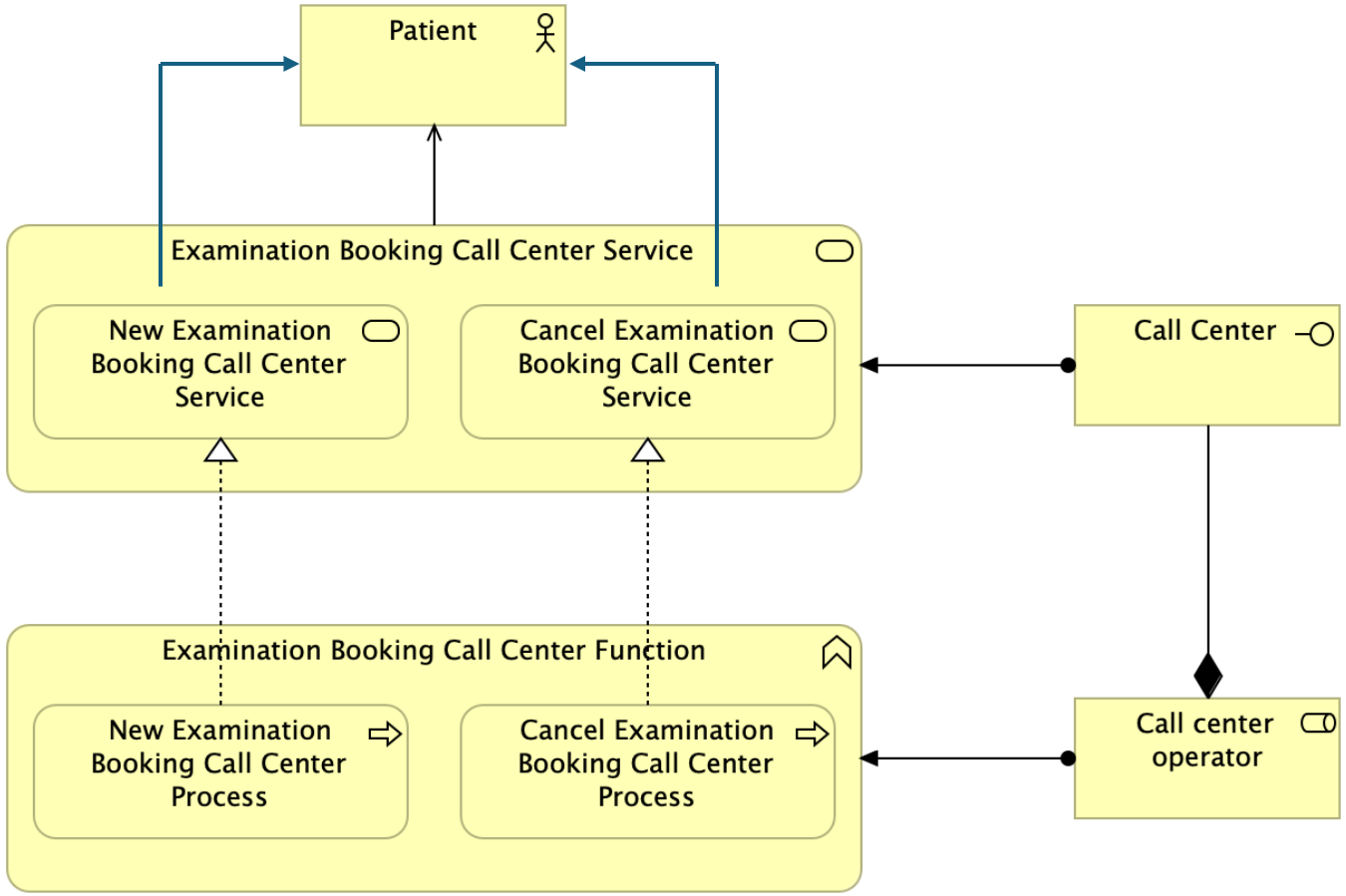
Suggestion: use nesting when the relation is “Composed of”, since nesting does not show the relation visually




Responsibility is propagated to nested elements

# Function specification

With nesting

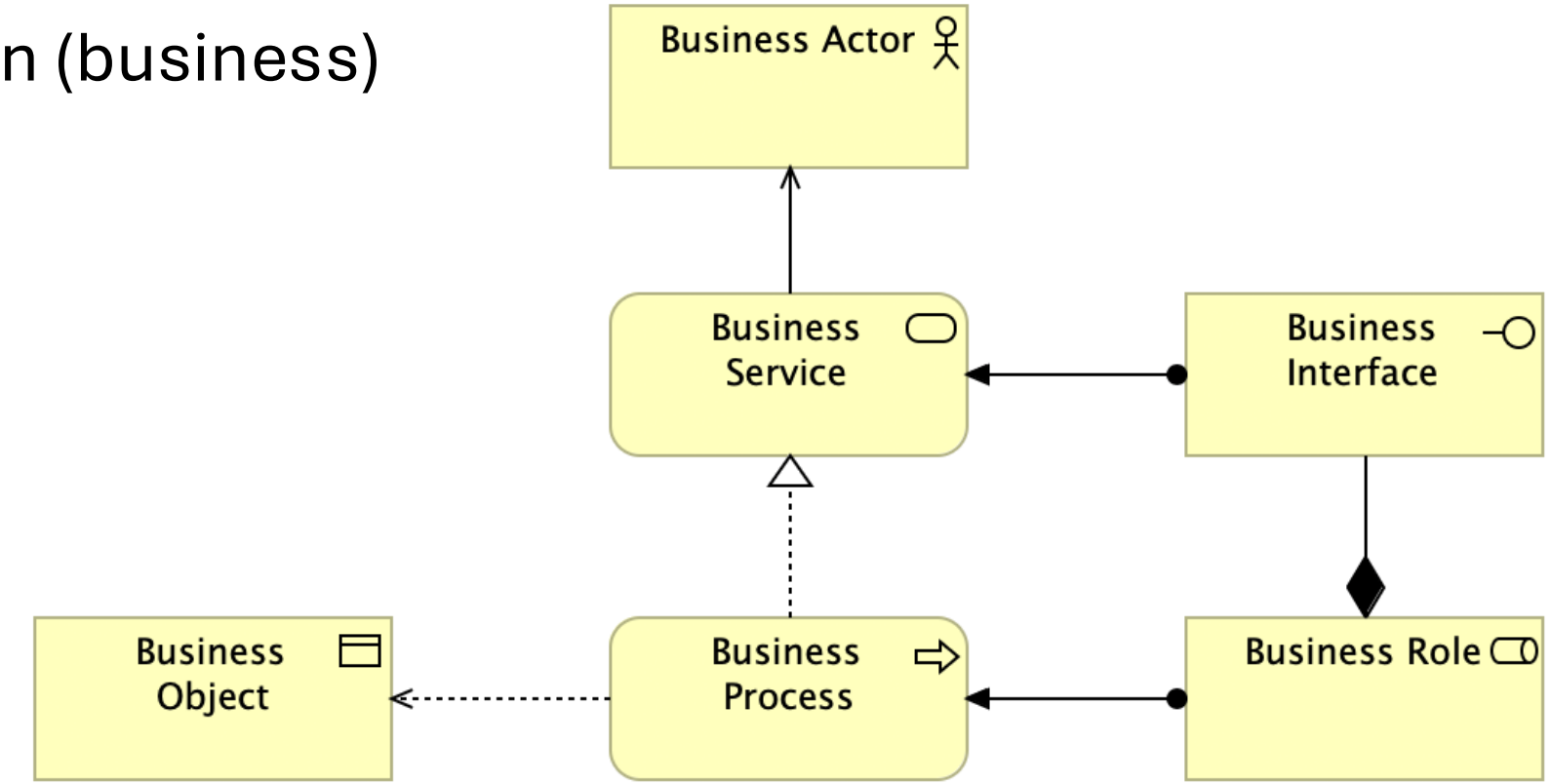



Nested elements in a function do not share a flow

# ArchiMate®
# Exercises with Archi

# ArchiMate

- Base pattern (business)



# Archi

- Speedy 01

Speedy is a delivery company that wants to offer to its top management a new service to create custom reports. The company already offers some services in this direction but the produced reports are not customizable. For this new service, a data analyst is available to the managers to understand their requirements and to develop the new report.

# Archi

- Speedy 02

To achieve this objective the data analyst will ask to complete a form, has to check the data available, and to develop the tools to produce the report which will be delivered to the manager

# Archi

- Speedy 03

Speedy is a delivery company that wants to create a new reporting service for the top management in addition to the reporting service already made available by the reporting function. The new service is offered via a Web interface which will allow the manager to select a geographical area first, then to analyze the different products and finally to create a report.

# Archi

- IC 01

  IC is an insurance company which wants to offer a new insurance service for small objects (< 2000$) managed completely online for reliable customers. The new process starts when new insurance request arrives.

  Then the customers credentials and past history are checked. If the customer is considered reliable, he can upload a photo of the item and its details (serial number, purchase date). At this point an employee estimates the item's price and, if < 2000$, the system sends a contract to the customers and waits for the signed copy. If the estimated price is > 2000$ the system sends a message to the customer explaining the service's conditions.

# Archi

- IC 02

  IC is an insurance company which wants to offer a new insurance service for small objects (< 2000$) managed completely online for reliable customers. The new process starts when new insurance request arrives.

  Then the customers credentials and past history are checked. If the customer is considered reliable, he can upload a photo of the item and its details (serial number, purchase date). At this point, an employee estimates the item's price and, if < 2000$, the customer and the legal representant start exchanging papers via e-mail to produce the signed contract. If the estimated price is > 2000$ the system sends a message to the customer explaining the service's conditions.