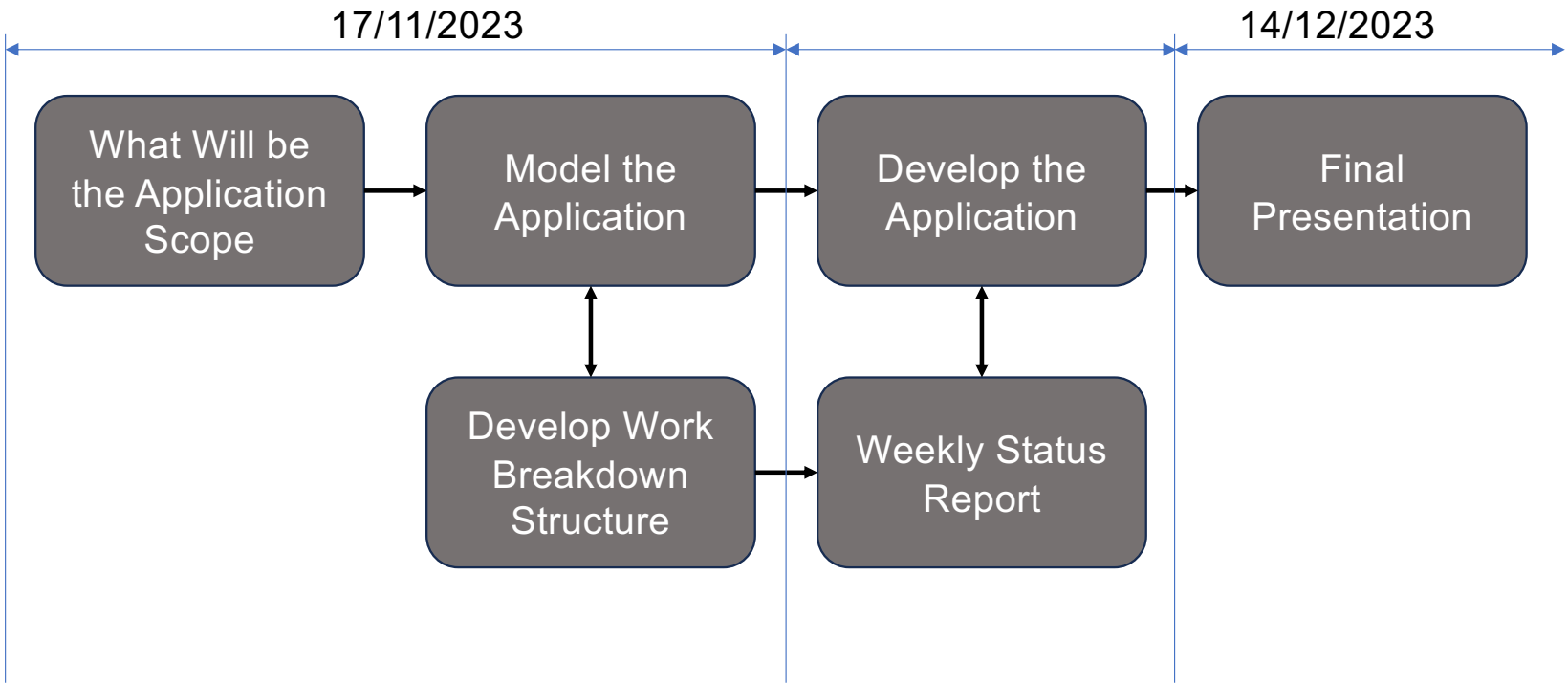

# Projecto Integrado 2

# Previous Class Deliverable Overview

# Project Structure



# Architecture

ISO/IEC 42010:

**Architecture**: the structure of a system in terms of

- its *components*,
- their *externally visible properties*,
- their *relations*,
- and the underlying *principles*

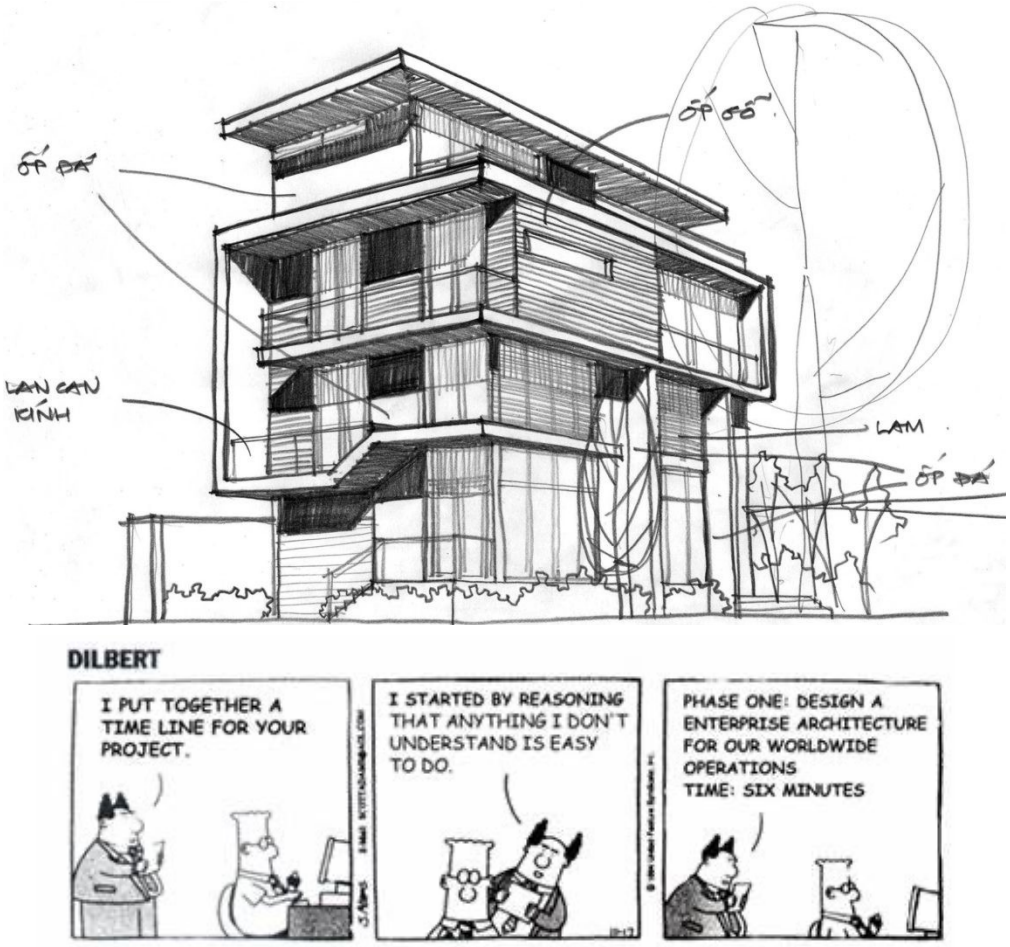
*“Structure with a vision”*

# Distinct Types of Architecture

| Architecture Type | Key Use Cases/Purposes |
|---|---|
| **Enterprise Architecture** | Enterprise Architecture (EA) is used to align business strategies, goals, and objectives with IT strategy, goals, and objectives. EA provides a holistic view of the organization's structure, processes, information, and technology. It helps organizations to optimize their operations, reduce costs, and improve efficiency by identifying opportunities for improvement and innovation. |
| **Infrastructure Architecture** | Infrastructure Architecture (IA) is used to design and manage the physical and logical components of an organization's IT infrastructure. IA includes the design of hardware, software, networks, data centers, and other components that support the organization's IT operations. IA helps organizations to ensure that their IT infrastructure is reliable, secure, and scalable. |
| **Software Architecture** | Software Architecture (SA) is used to design and develop software systems that meet the organization's business requirements. SA includes the design of software components, modules, interfaces, and other elements that make up the software system. SA helps organizations to ensure that their software systems are scalable, maintainable, and extensible. |
| **Cloud Architecture** | Cloud Architecture (CA) is used to design and manage cloud-based solutions that meet the organization's business requirements. CA includes the design of cloud infrastructure, platforms, and applications. CA helps organizations to leverage the benefits of cloud computing, such as scalability, flexibility, and cost-effectiveness. |

# Why Enterprise Architecture?

- Managing change and complexity:
  - Aligning business and IT
  - Outsourcing
  - Impact analysis
  - Project support (project start architectures)
  - Portfolio management
  - Communication with stakeholders ...
- Obtaining insight in current situation (as-is)
- So enterprise architecture as a tool
  - for communication
  - for governance
  - for innovation



ipt.pt

# Key Concepts

- A ***stakeholder*** is a person or organization with a certain interest in (part of) an architecture
- A ***view*** is a representation of a system from the perspective of a set of concerns of one or more stakeholders. A view is what you see.
- A ***viewpoint*** is where you are looking from. It defines how to build a view, e.g. by means of a template.

- Different stakeholders
  - Have different interests and use different concepts
  - Have different views
  - Have different viewpoints
  - On the basis of one consistent architectural model

# Example viewpoint en view

**Example 1**

- Stakeholder: city council
- Concern: development plan
- Viewpoint: photo of district from the air
- View: Aerial

**Example 2**

- Stakeholder: house owner
- Concern: building permit
- Viewpoint: Corner of the street, scope is one house
- View: Ground photo

# K&C Application - Stakeholder

In the context of the kennel and cattery management application, the following stakeholders can be identified:

**Anonymous/Public**: This stakeholder group is interested in browsing through the animal inventory to see which animals are available for adoption. They are concerned with the ease of use and accessibility of the application.

**Kennel Support Personnel**: This stakeholder group is interested in taking pictures of the animals and maintaining their status/update with details about them. They are concerned with the usability and functionality of the application.

**Kennel Backoffice Secretary**: This stakeholder group is interested in treating the animal adoption process. They are concerned with the accuracy and completeness of the information in the application.

**Kennel Manager**: This stakeholder group is interested in seeing the overall status of the animal inventory. They are also concerned with the usability of the application and its impact on daily activities (the impact of having to register data in an application for a non-tech-savvy population).

# K&C Application - Viewpoints

To model the architecture of the kennel and cattery management application, the following views and viewpoints can be used:

**Business Viewpoint**: This viewpoint describes the business processes and organizational structures of the kennel and cattery. It includes the goals, objectives, and processes of the organization, as well as the roles and responsibilities of the stakeholders involved.

**Application Viewpoint**: This viewpoint describes the software applications that support the business processes of the kennel and cattery. It includes the design of the software components, modules, interfaces, and other elements that make up the software system.

**Information Viewpoint**: This viewpoint describes the information flows and data structures of the kennel and cattery. It includes the design of the data models, databases, and other information systems that support the business processes of the organization.

**Technology Viewpoint**: This viewpoint describes the hardware and software infrastructure that supports the applications of the kennel and cattery. It includes the design of the networks, servers, storage devices, and other components that make up the IT infrastructure.

# K&C Application - Views

To model the architecture of the kennel and cattery management application, the following views and viewpoints can be used:

**Stakeholder View:** This view describes the stakeholders involved in the kennel and cattery management application, their interests, and their requirements. It includes the anonymous/public, kennel support personnel, kennel backoffice secretary, and kennel manager.

**Business Process View:** This view describes the business processes of the kennel and cattery management application. It includes the processes for browsing through the animals' inventory, taking pictures of the animals, maintaining their status/update with details about them, treating the animal adoption process, and seeing the overall status of the animal's inventory.

**Application View:** This view describes the software applications that support the business processes of the kennel and cattery management application. It includes the design of the software components, modules, interfaces, and other elements that make up the software system.

**Information View:** This view describes the information flows and data structures of the kennel and cattery management application. It includes the design of the data models, databases, and other information systems that support the business processes of the organization.

**Technology View:** This view describes the hardware and software infrastructure that supports the applications of the kennel and cattery management application. It includes the design of the networks, servers, storage devices, and other components that make up the IT infrastructure.

# What is Modelling

**Modelling is the process of creating a simplified representation of a complex system or concept.** It involves the use of abstract or mathematical models to assist calculations and predictions. Modelling can be used in various fields such as engineering, economics, and computer science, to name a few. The benefits of modelling include reducing development risks, improving communication, helping visualize the end product, and facilitating iterative development.

# Why Modelling?

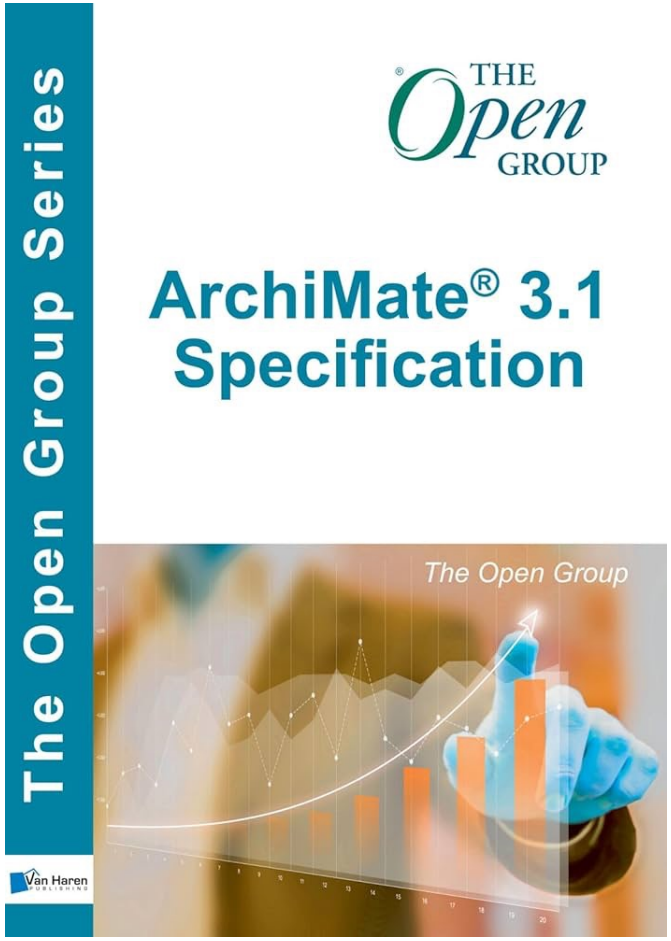
- **Reduces Development Risks:** Modelling helps identify potential issues and risks early on in the development process, which can save time and money in the long run.
- **Improves Communication:** Models provide a common language for all stakeholders involved in the development process, which can help improve communication and collaboration.
- **Helps Visualize the End Product:** Models can help stakeholders visualize the end product and ensure that it meets their requirements and expectations.
- **Facilitates Iterative Development:** Models can be used to test and refine ideas before committing to a final design, which can help facilitate an iterative development process2.

# Modelling K&C Management

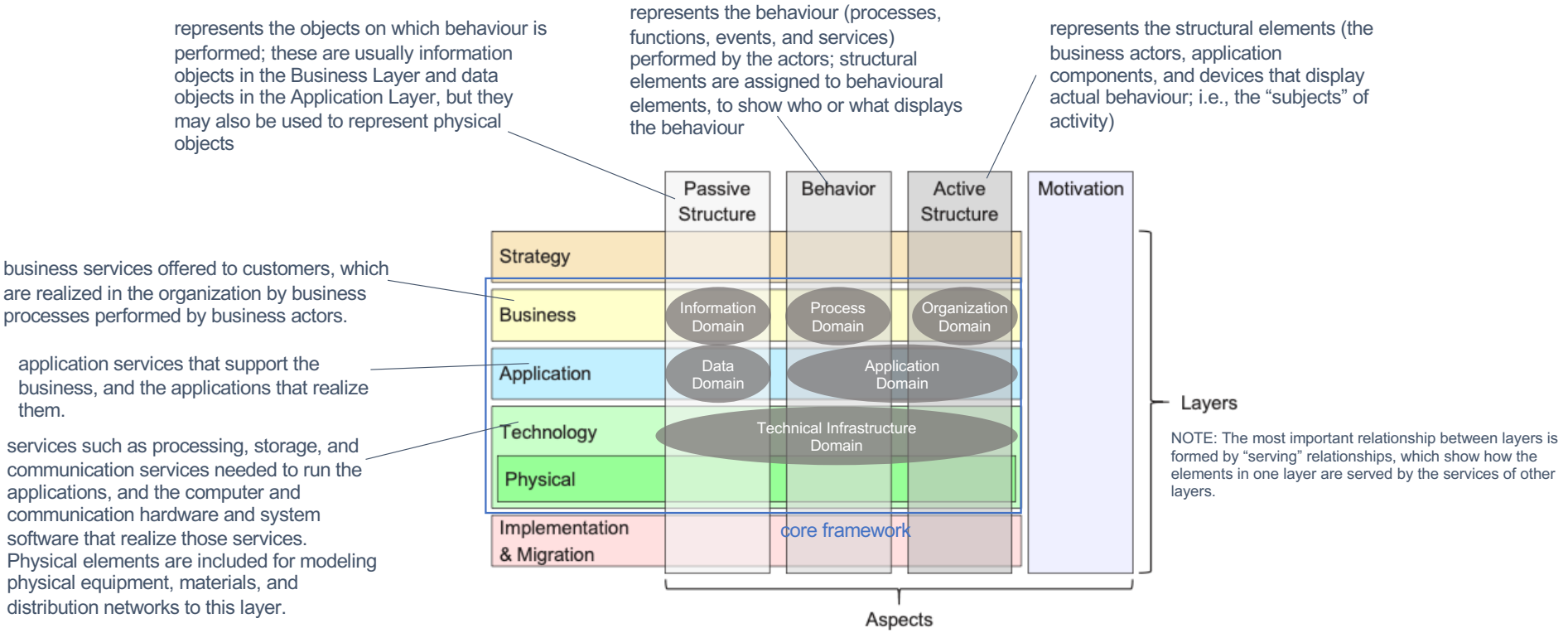
In the context of kennel and cattery management application, **modelling** refers to the process of creating a simplified representation of the system or concept that you want to develop. In this case, it would involve creating a model of the kennel and cattery management application that you want to build. The model would help you identify potential issues and risks early on in the development process, provide a common language for all stakeholders involved in the development process, help stakeholders visualize the end product and ensure that it meets their requirements and expectations, and facilitate an iterative development process

# ArchiMate

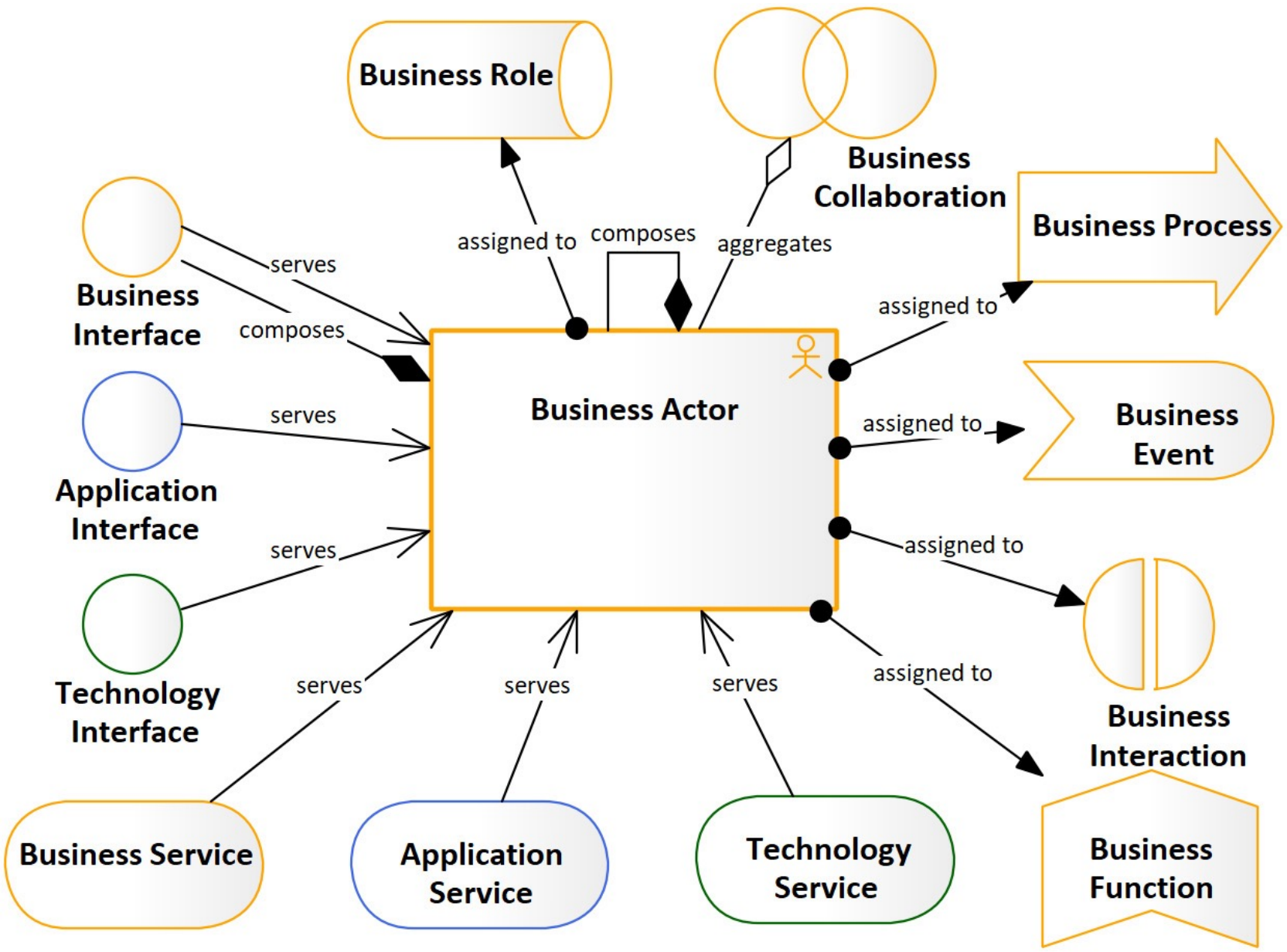
- A language for describing architectures
- Covers business, application and technology layers
  - With relations between these layers
- Graphical language with formal semantics, enabling analysis and tool support
- Techniques for visualization and analysis, aimed at various stakeholders
- Open standard maintained by The Open Group
- See www.opengroup.org/archimate or www.archimate.org

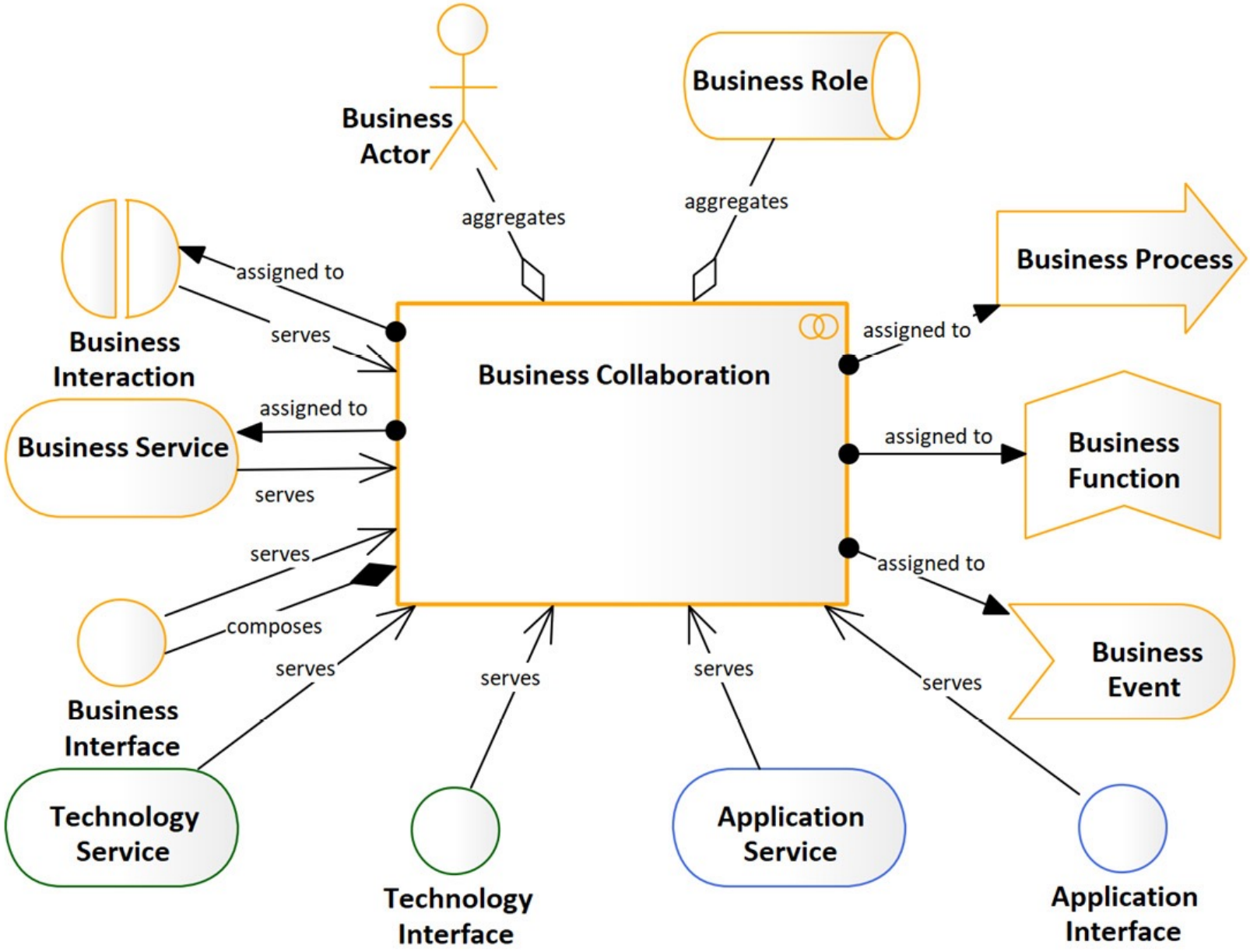


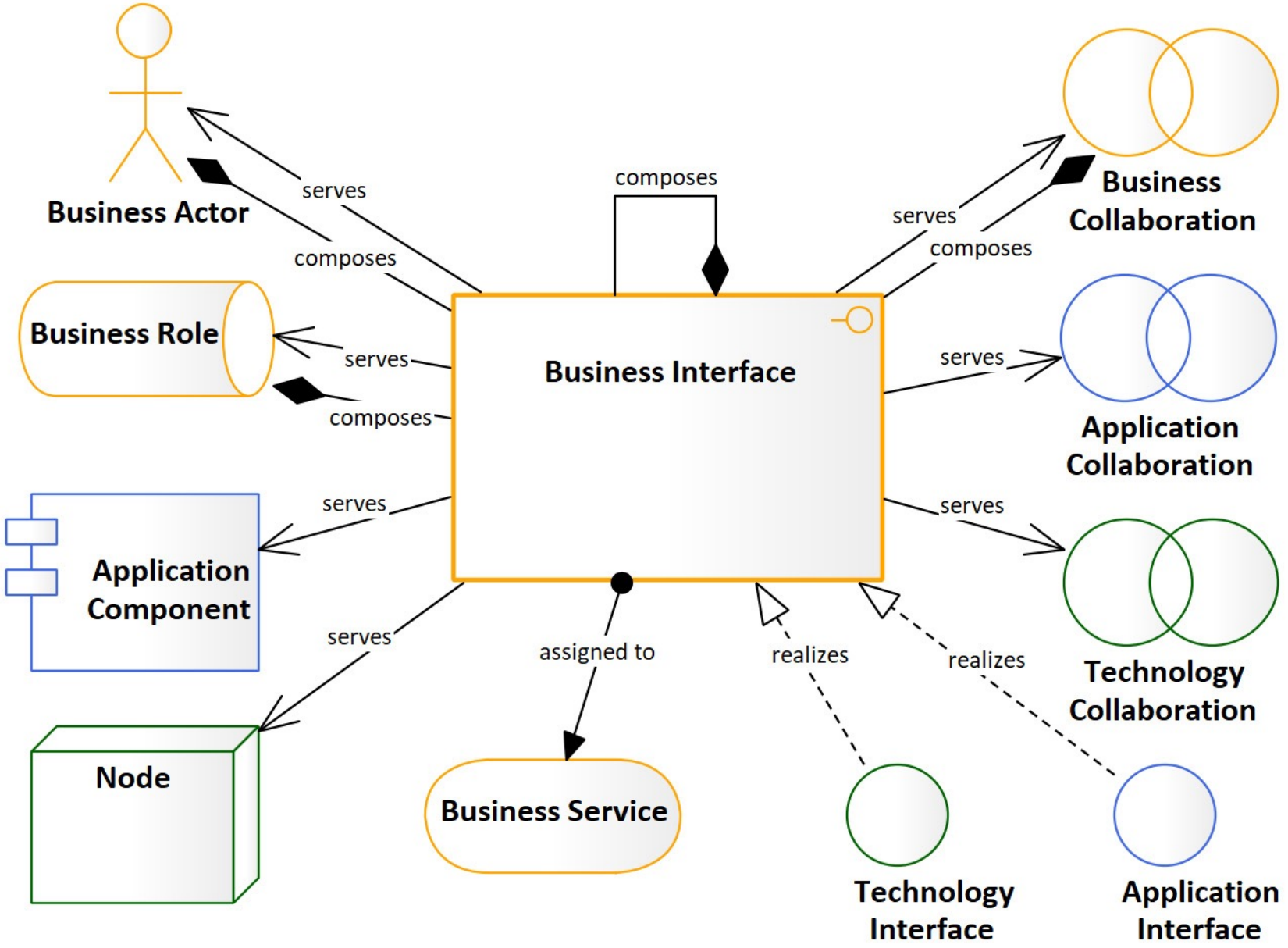
# ArchiMate Modelling Language

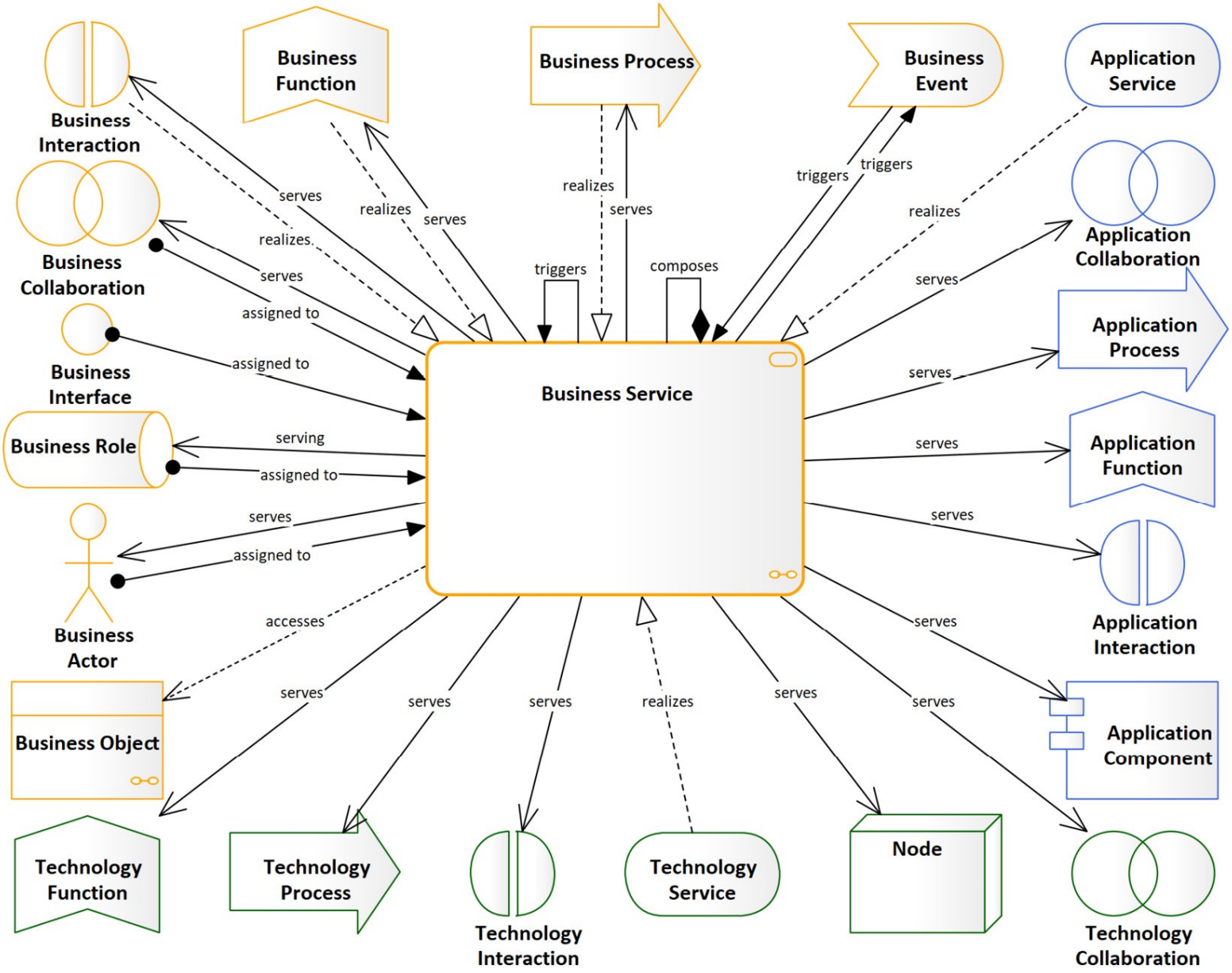



**it is often useful to start with the behaviors that the system must perform, while in modeling existing systems, it is often useful to start with the people, applications, and infrastructure that comprise the system, and then analyze in detail the behaviors performed by these active structures.**

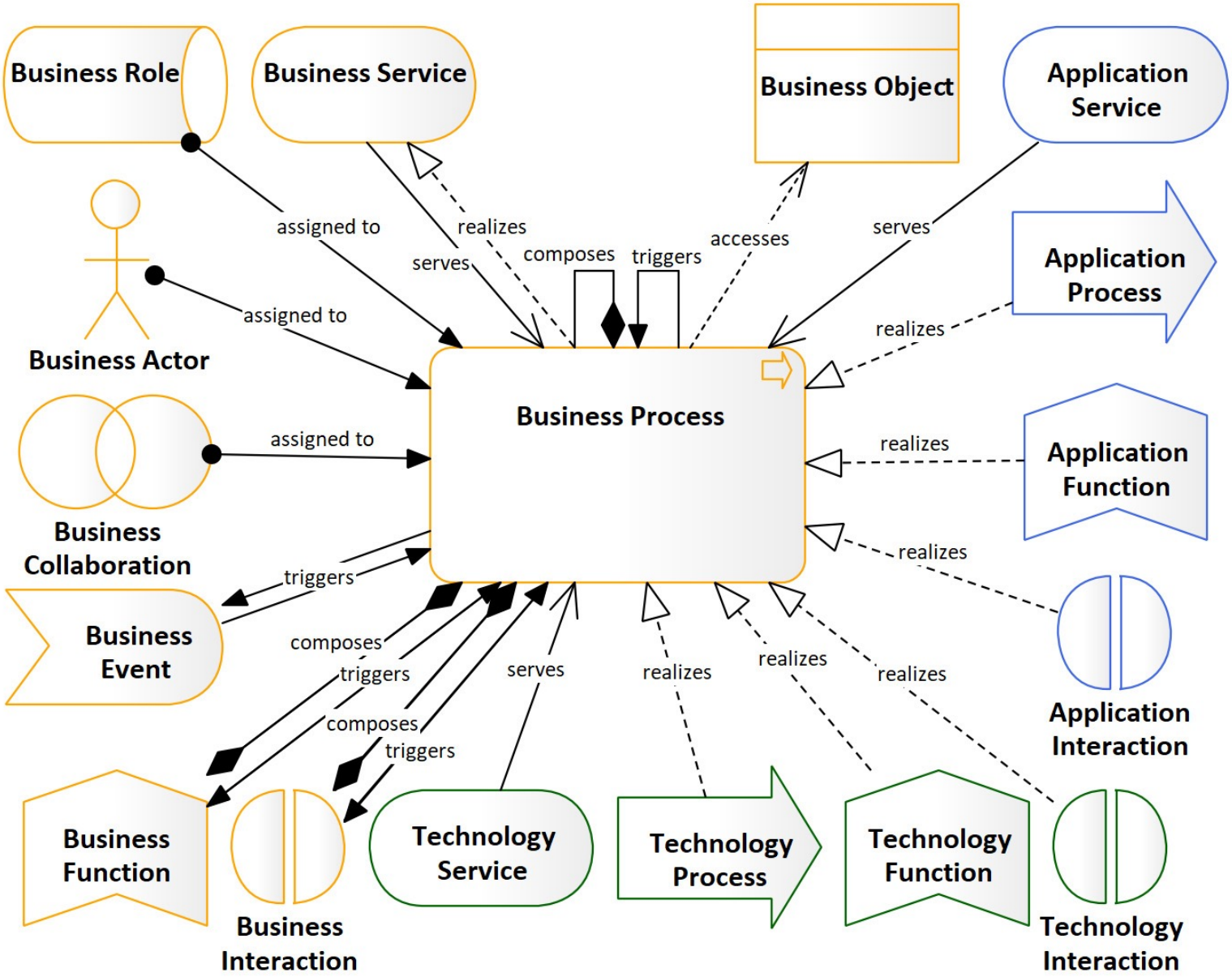

Business Role
Business Collaboration
Business Process
assigned to
composes
aggregates
Business Interface
serves
composes
assigned to
Business Actor
Application Interface
serves
assigned to
Business Event
serves
assigned to
Technology Interface
serves
serves
serves
assigned to
Business Interaction
Business Service
Application Service
Technology Service
Business Function


Business Actor
Business Role
aggregates
aggregates
Business Process
assigned to
Business Interaction
serves
Business Collaboration
assigned to
assigned to
Business Service
Business Function
serves
serves
assigned to
composes
Business Event
Business Interface
serves
serves
serves
serves
Technology Service
Application Service
Technology Interface
Application Interface

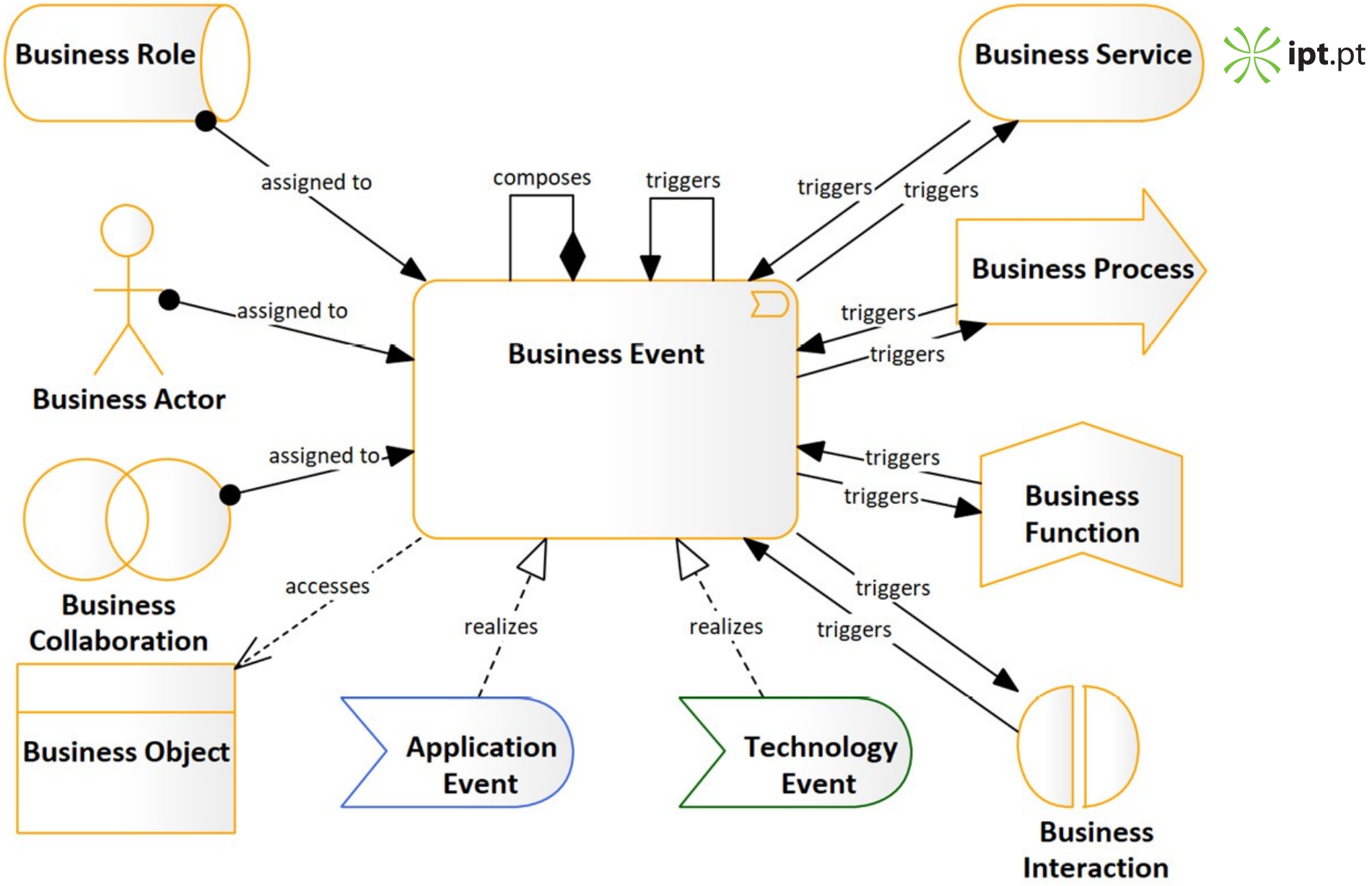

Business Actor
serves
composes
Business Role
serves
composes
Application Component
serves
serves
Node
composes
Business Interface
assigned to
Business Service
serves
composes
Business Collaboration
serves
Application Collaboration
serves
Technology Collaboration
realizes
Technology Interface
realizes
Application Interface


Business Interaction
Business Function
Business Process
Business Event
Application Service
Business Collaboration
Application Collaboration
Business Interface
Application Process
Business Service
Business Role
Application Function
Business Actor
Application Interaction
Business Object
Application Component
Technology Function
Technology Process
Technology Interaction
Technology Service
Node
Technology Collaboration
serves
realizes
assigned to
triggers
composes
serving
accesses

ipt.pt


Business Role
Business Service
Business Object
Application Service
Business Actor
Application Process
Business Process
Application Function
Business Collaboration
Business Event
Application Interaction
Business Function
Business Interaction
Technology Service
Technology Process
Technology Function
Technology Interaction
assigned to
realizes
serves
composes
triggers
accesses
assigned to
assigned to
realizes
realizes
realizes
triggers
composes
triggers
composes
triggers
serves
realizes
realizes
realizes

ipt.pt


Business Role
Business Service
ipt.pt
assigned to
composes
triggers
triggers
triggers
Business Process
assigned to
Business Actor
Business Event
triggers
triggers
assigned to
triggers
triggers
Business Function
Business Collaboration
accesses
realizes
realizes
triggers
triggers
Business Object
Application Event
Technology Event
Business Interaction