



midwest  
architecture community  
collaboration

ASA

## Archi Overview

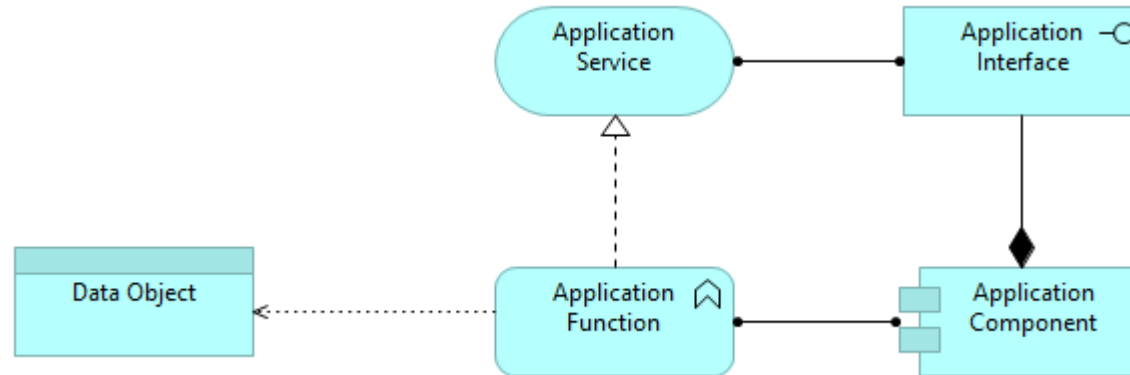
# ArchiMate

- ArchiMate is built from three types of elements:
  - elements that act (active elements)
  - elements that represent the behavior of those 'elements that act' (behavioral elements)
  - elements that cannot act and which are acted upon by that behavior (passive elements)
- The three element types, connected by relations, can form sentences of sorts.
- The structure of the ArchiMate grammar is partly based on the subject-verb-object pattern from natural language.
- It tells the reader the basic structure of the story: who acts on what.

# ArchiMate

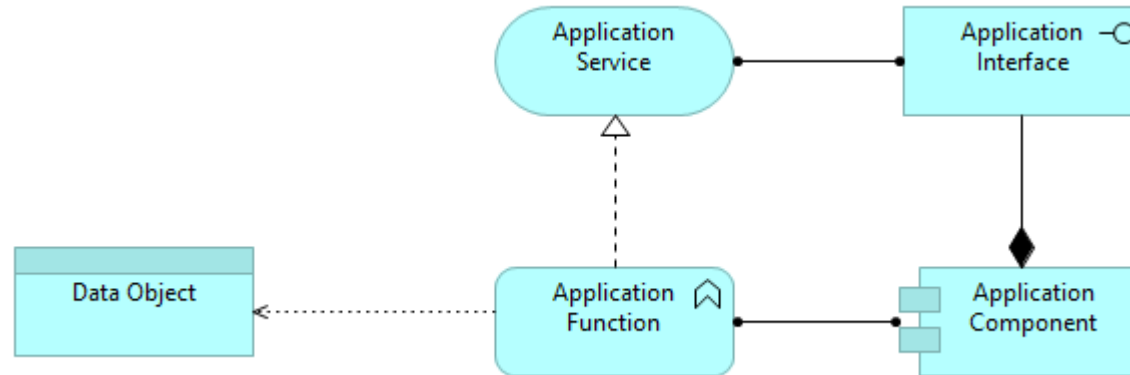
- Divides Enterprise Architecture into a Business & Information layer, an Application & Data layer and a Technical Infrastructure layer. These are the rows in ArchiMate's meta-model and this division is fairly standard in the Enterprise Architecture world.
- Divides architecture in any layer 'strictly' into Active Structure, Behavior and Passive Structure. Put in a sentence: "Who/what does what to what?". The clear separation of actors and their behavior is not common in Enterprise Architecture and it is a main foundational aspect of ArchiMate.
- Has three kinds of relations: Structural Relations, Dynamic Relations and 'Other' Relations.

# Basic Application Pattern



- **Application Component**
  - Stands for the 'actor' that an application in your Enterprise Architecture landscape is
- **Application Function**
  - Stands for the behavior of the Application Component, how the application can act. It is one side of the coin of which Application Component is the other.
- **Data Object**
  - Is what the Application Function acts upon. The Application Function might create, read, write, update or delete the Data Object.
  - Only model (semi) persistent data.

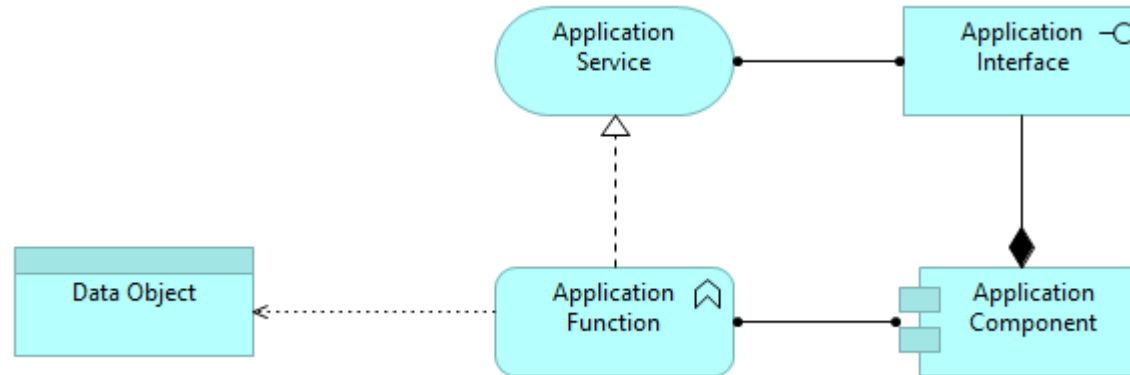
# Basic Application Pattern



They have to do with how the application is used/seen (by the business or by other applications):

- **Application Interface**
  - Stands for the route via which the application offers itself to the business or to other applications.
  - both separate concepts (used by people and used by other applications) are supported by this one element.
  - Application Interface is a 'handle' of the 'actor' that is the Application Component. It is one side of a coin of which Application Service is the other.
- **Application Service**
  - Stands for the 'visible' behavior of the Application, how the Application Interface can act for a user.
  - It is one side of the coin of which Application Interface is the other side.

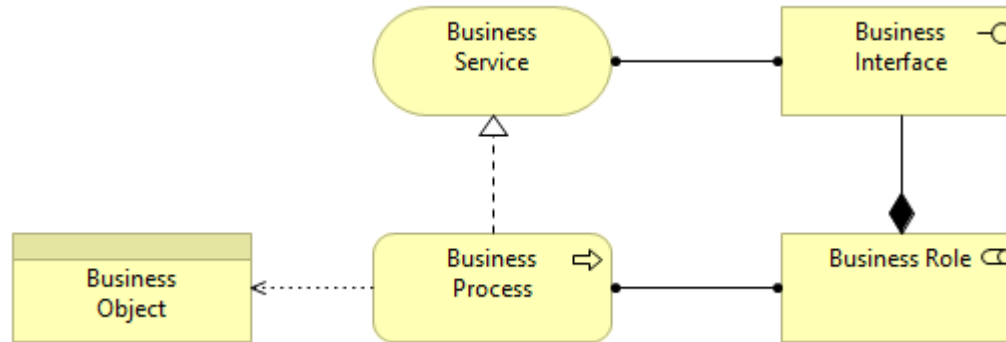
# Basic Application Pattern



## Relations:

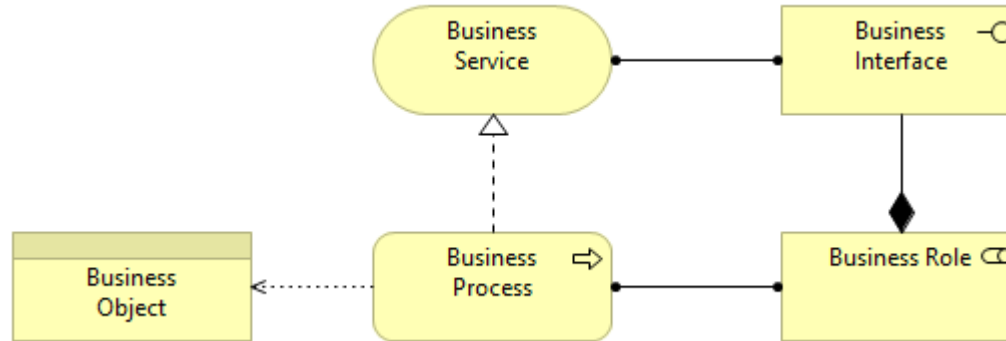
- Access Relation
  - The access relation always depicts a behavioral element accessing a passive element.
- Composition Relation
  - It means that the element at the end with the diamond is the parent of the element on the other end and that the child cannot exist independently from the parent.
- Realization Relation
  - Application's internal functionality realizes a service, which is the externally usable functionality of the application.
- Assignment Relation
  - One side (the active element) performs the behavior that is the behavioral element on the other side.

# business & information



- **Business Role**
  - The Business Role is an 'actor' in ArchiMate.
  - Business Roles can perform Business Processes.
- **Business Process**
  - Stands for a set of causally-related activities that together realize services or create elements.
  - Roles can be assigned to a Business Process, they perform the process.
- **Business Object**
  - Generally the abstract element that is created or used by a Business Process.

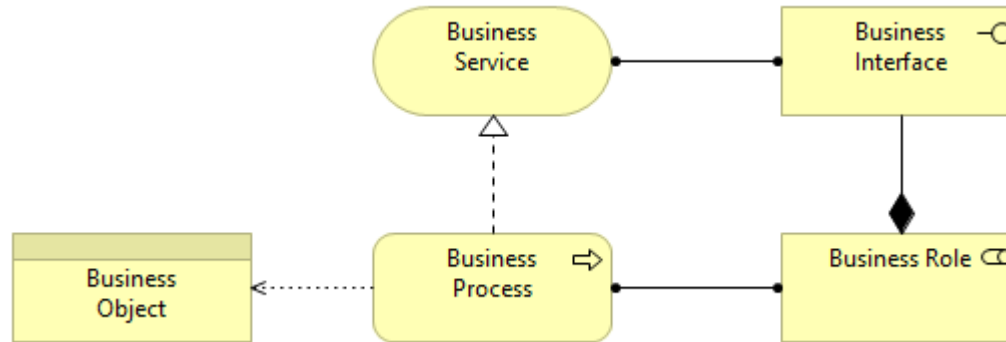
# business & information



- **Business Interface**
  - The way the role interacts with others.
  - The interface is the visible manifestation of a role.
- **Business Service**
  - This is the reason for the existence of the process.
  - This is the service it offers to (and thus can be used by) others (either inside or outside the company).
  - A company might offer many services.



# business & information



- **Business Interface**
  - The way the role interacts with others.
  - The interface is the visible manifestation of a role.
- **Business Service**
  - This is the reason for the existence of the process.
  - This is the service it offers to (and thus can be used by) others (either inside or outside the company).
  - A company might offer many services.

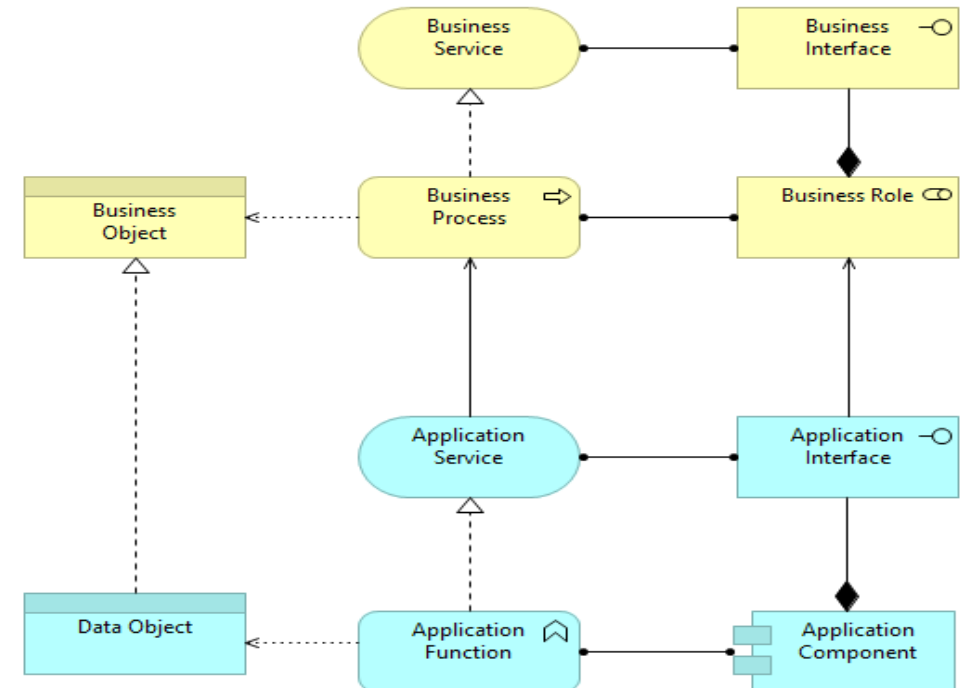
# Linking Levels

The application level is connected to the business level by three relations.

On the left we see the already familiar Realization relation. Here it means that the Data Object realizes the Business Object.

In the middle and on the right we see a new relation: This is the Used-By relation. It means that the element at the end without the arrowhead is used by the element at the end with the arrow head.

The Application Service, for instance, is Used-By the Business Process. The Application Interface (e.g. the Graphical User Interface) is Used-By the Business Role.



# Business Process and Business Function

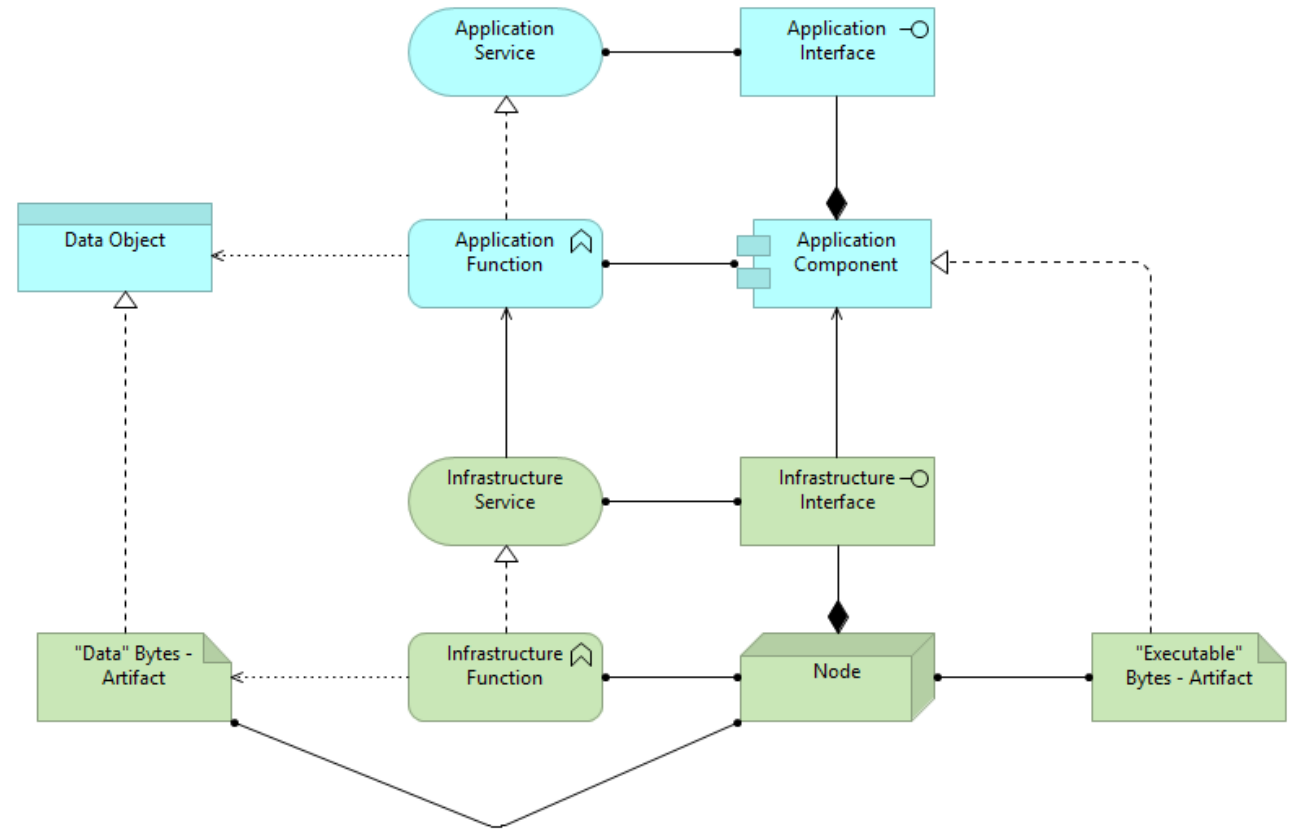
- You use a Business Process if you are thinking of a causally-related set of behaviors ('activities') that in the end produce something, normally a service or an object.
- You use a Business Function if you are thinking of a grouping of related behavior based on — for instance — same tools, same skills or same role that performs it. In fact, Business Function is best seen as an inside-out view of the business behavior.

# Business Actor

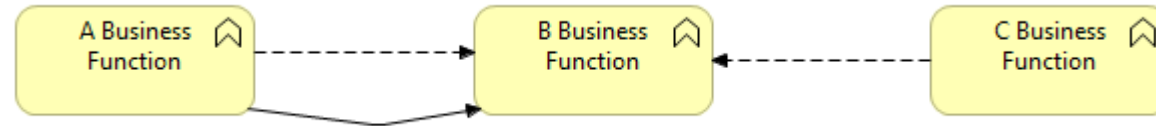
- The Business Role is an abstract sort of actor.
- The Business Actor is a “real” actor. People, departments, business units, companies regulators etc.

# Technical Infrastructure

- **Node**
  - Hardware and system software where files are stored or applications can run.
- **Infrastructure Function**
  - Behavior of the Node
- **Infrastructure Service**
  - Visible behavior of the Node
  - What the applications need to function.
    - File share
    - Application execution
    - Database Service
- **Infrastructure Interface**
  - Contract that the application has to fulfill.
    - SMB, NFS, TCP/IP etc.
- **Artifact**
  - File, Database, Executable file



# Event, Trigger and Flow

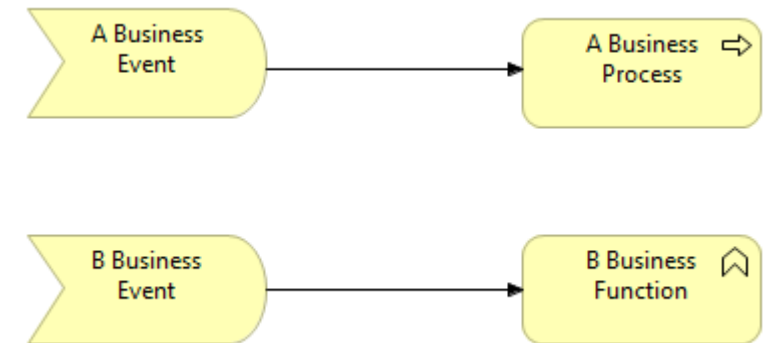


- **Trigger**
  - B receives order from A to start “trigger” function.
  - Triggering means there is a causal relation between the two functions.
- **Flow**
  - Information flows from one to the other.

# Event, Trigger and Flow

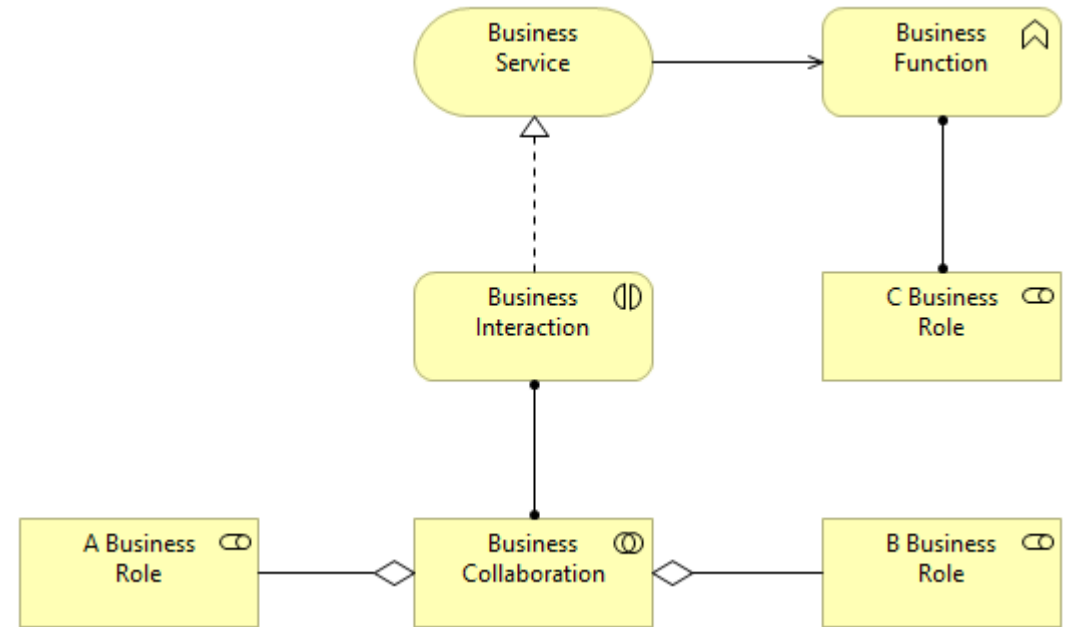
## ■ Business Event

- Is 'something that happens'.
- Events can trigger a Business Process or a Business Function and they can be raised by a Business Process or Business Function, which is also depicted with a Trigger relation.
- Business Events are normally used for standalone 'things that happen'.



# Business Collaboration

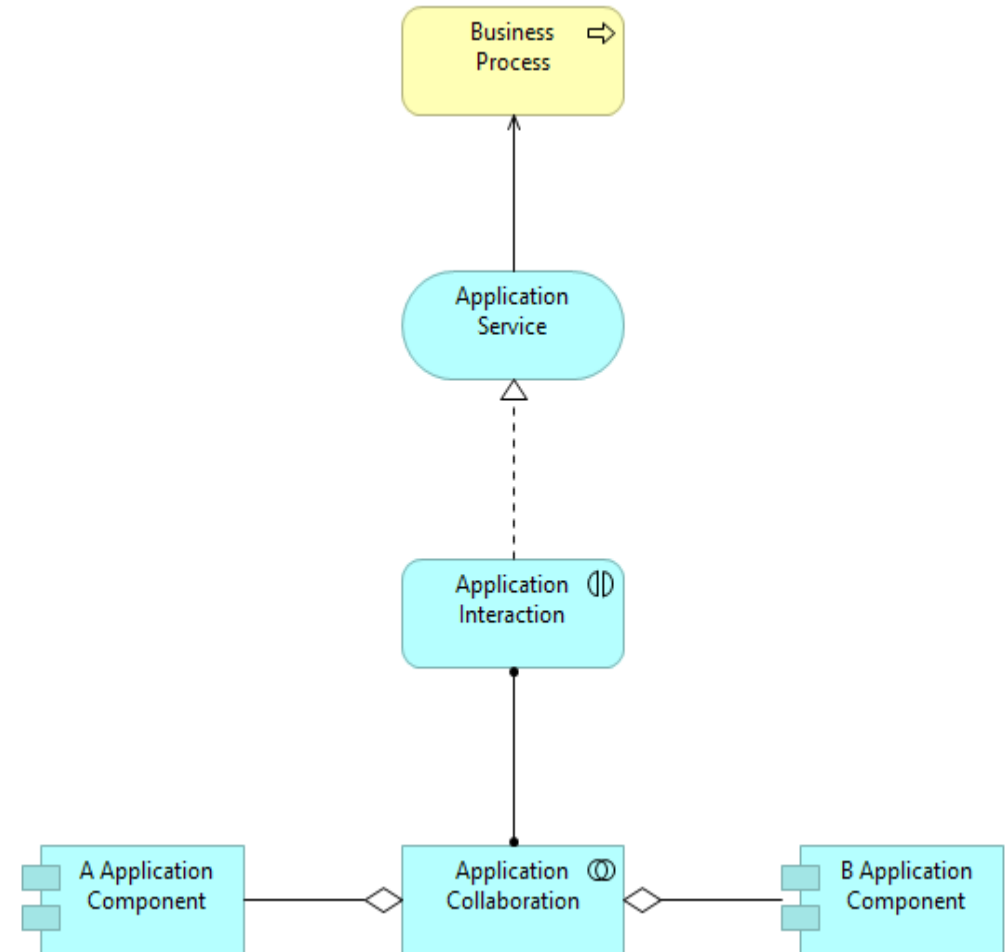
- **Business Collaboration**
  - Two or more roles collaborating to provide a business interaction (behavior)
  - All parties own the business interaction
  - Not clear who is in charge
- **Business Interaction**
  - Behavior of the Business Collaboration





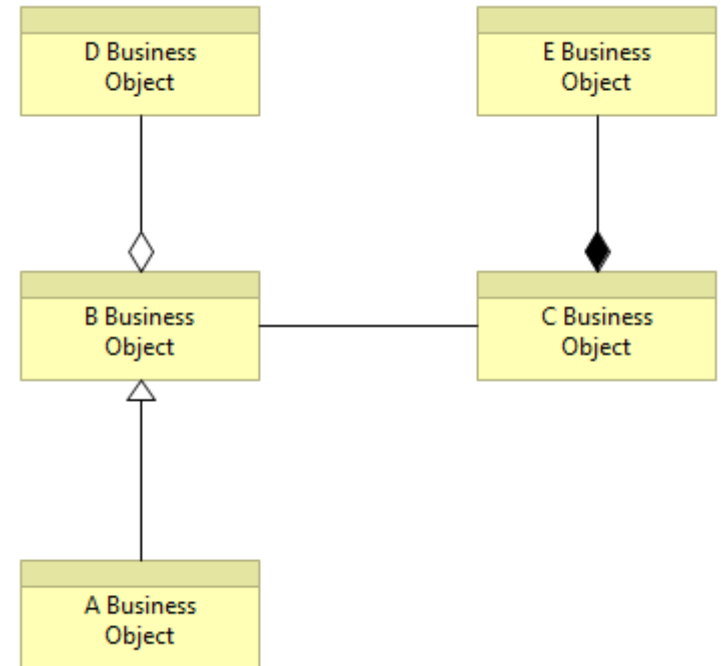
# Application Collaboration

- **Application Collaboration**
  - Two or more components collaborating to provide an **Application Interaction**
  - Not clear which component is in charge
- **Application Interaction**
  - Behavior of the **Application Collaboration**



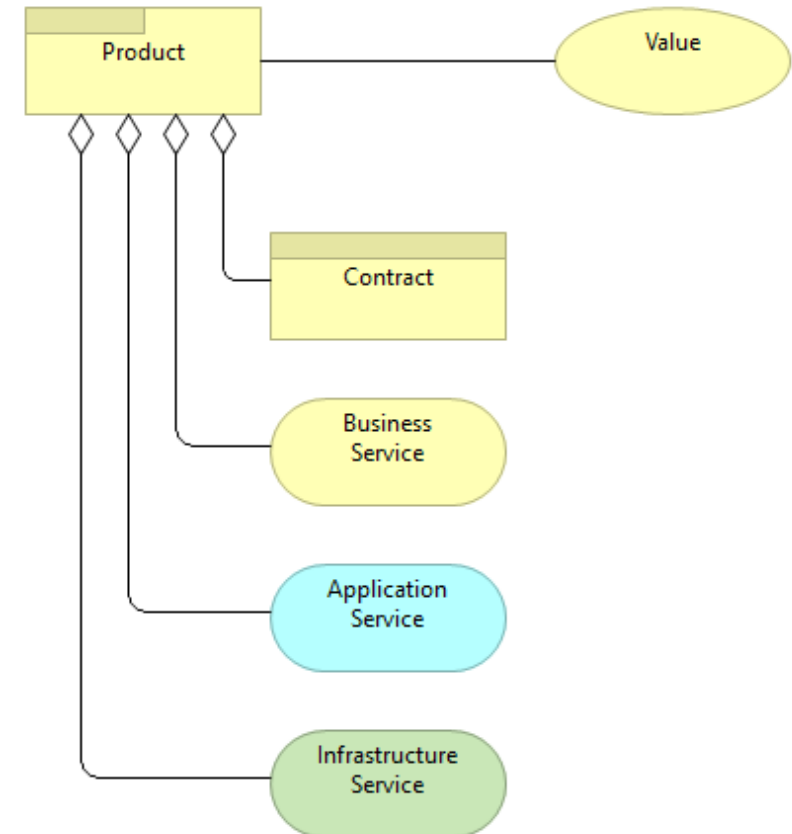
# Relations

- **Association Relation**
  - Catch-all relation
- **Specialization Relation**
  - Specialization of another element
  - Is-A
  - Support Multiple inheritance
- **Composition Relation**
  - Has-A
  - Refers-To



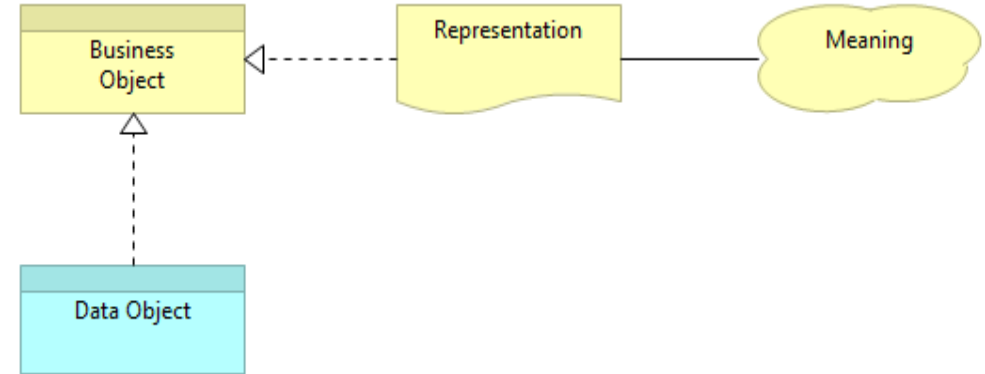
# Product, Contract and Value

- **Product**
  - What you offer to the outside world
    - Could be outside business or outside department etc.
  - Aggregation of one or more services and optionally a contract
- **Contract**
  - Formal or informal agreement that covers the delivery of the service provided
- **Value**
  - Value of the Product
  - Abstract
    - Value to consumer
    - Value to producer
    - Be described in monetary or emotional terms
  - Officially it is associated with Service, and only indirect with Product (as opposed to here)



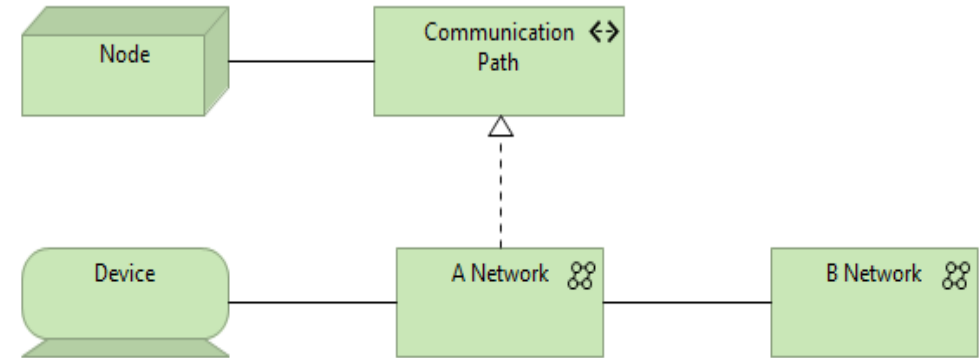
# Representation and Meaning

- **Representation**
  - Representation that can be shared with others, e.g. a print, a PDF etc.
- **Meaning**
  - The meaning of the representation
  - The information related counterpart of a Value
  - Represents the intention of a Business Object or Representation



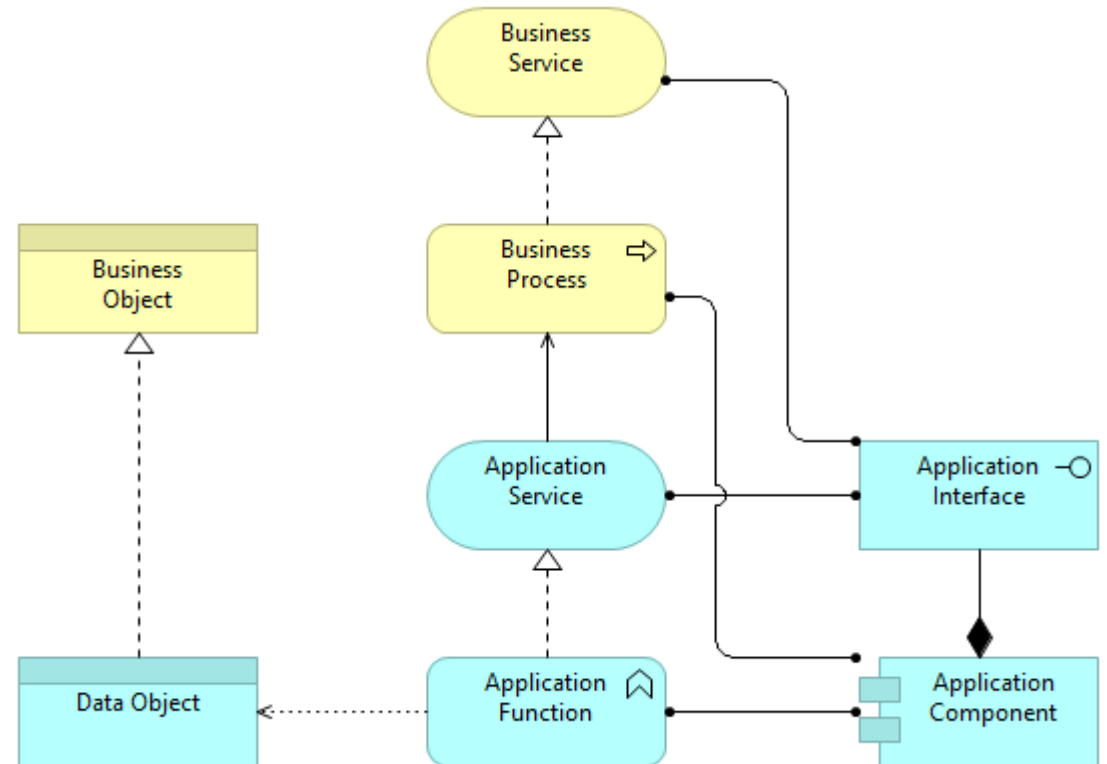
# Network and Communication Path

- **Network**
  - Physical communication medium
  - Realizes a Communication Path
- **Communication Path**
  - A way for nodes to exchange data
  - The logical representation of a physical Network



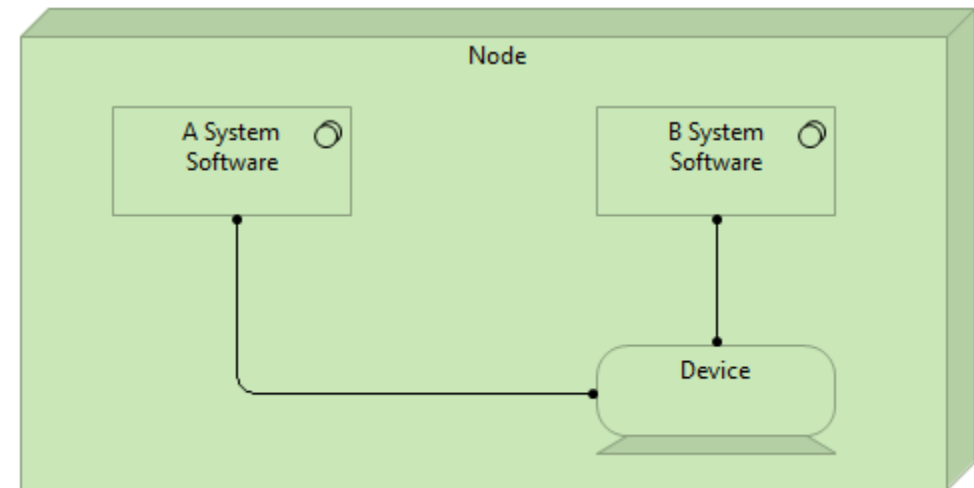
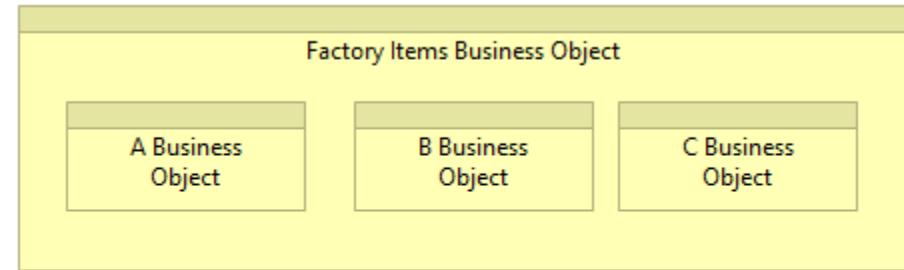
# Automated Processes

- If a process is run by people, a Business Role is Assigned-To the Business Process and a Business Interface is Assigned-To a Business Service.
- If a Business Process is performed by an application we draw an Assigned-To between the Application component and the Business Process and we also draw an Assigned-To between the Application Interface and the Business Service



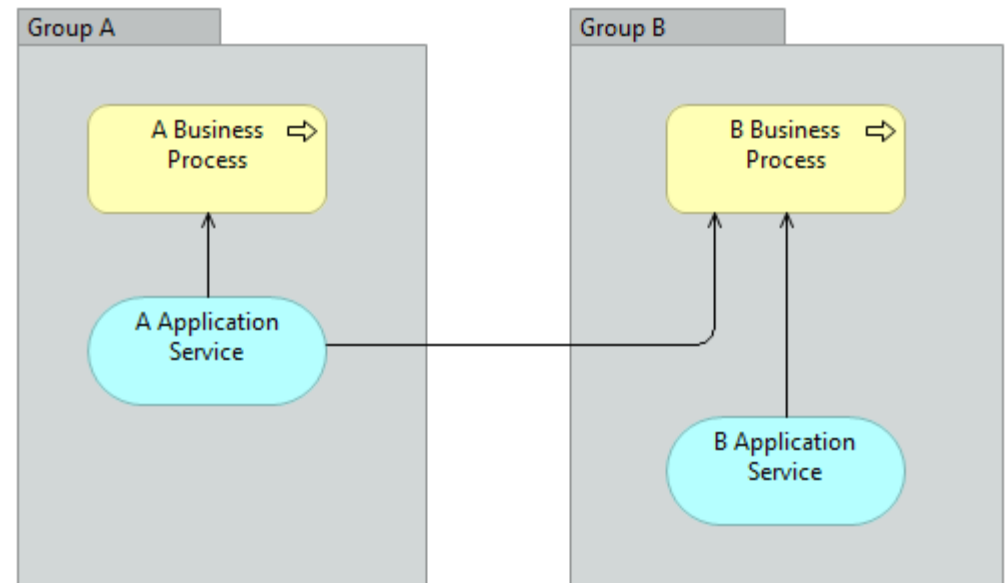
# Nesting

- There are three relation types that may be drawn by nesting an element inside another element: Composition, Aggregation and Assignment.
- Archi makes you select one relationship type, but you can't see it visually. A and B is Composition and C is Aggregation but looks the same.
- You can use Node to encapsulate Infrastructure.



# Grouping

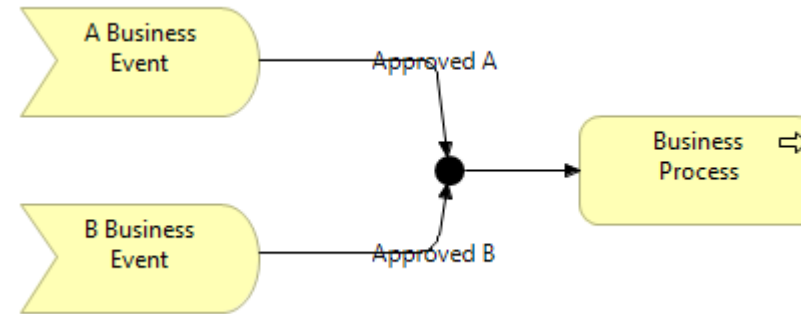
- Nesting (described in prior slide)
- Use Aggregation to group elements to a common “base” element.
- Grouping Relation
  - Catch-all relation





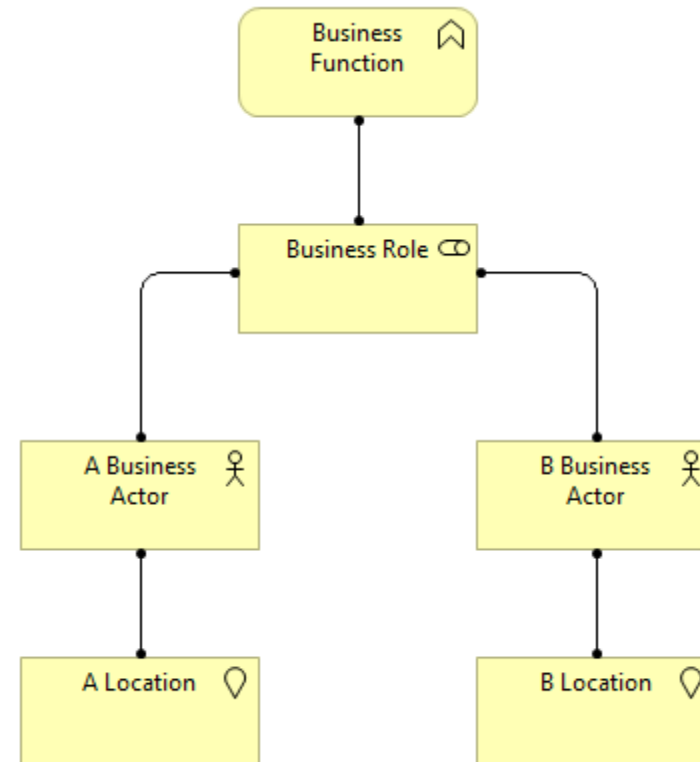
# Junction Relation

- Relate other relations to each other



# Location

- Geographical location where something resides.
- You can use the Assignment relation to link it to several other element types to model where they are located.



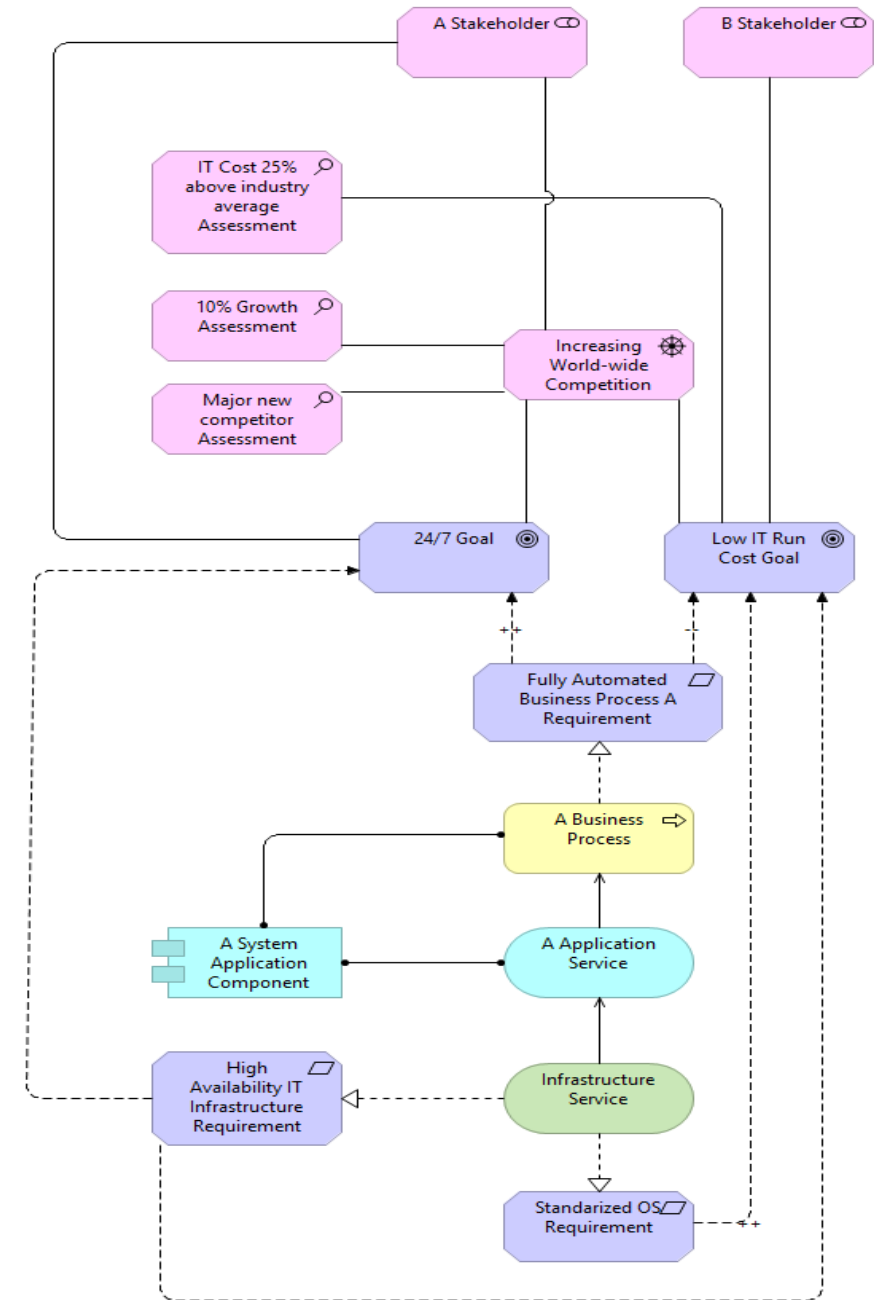
# Archi Grouping and Notes Icons

- Note
- Grouping
- Note Connection



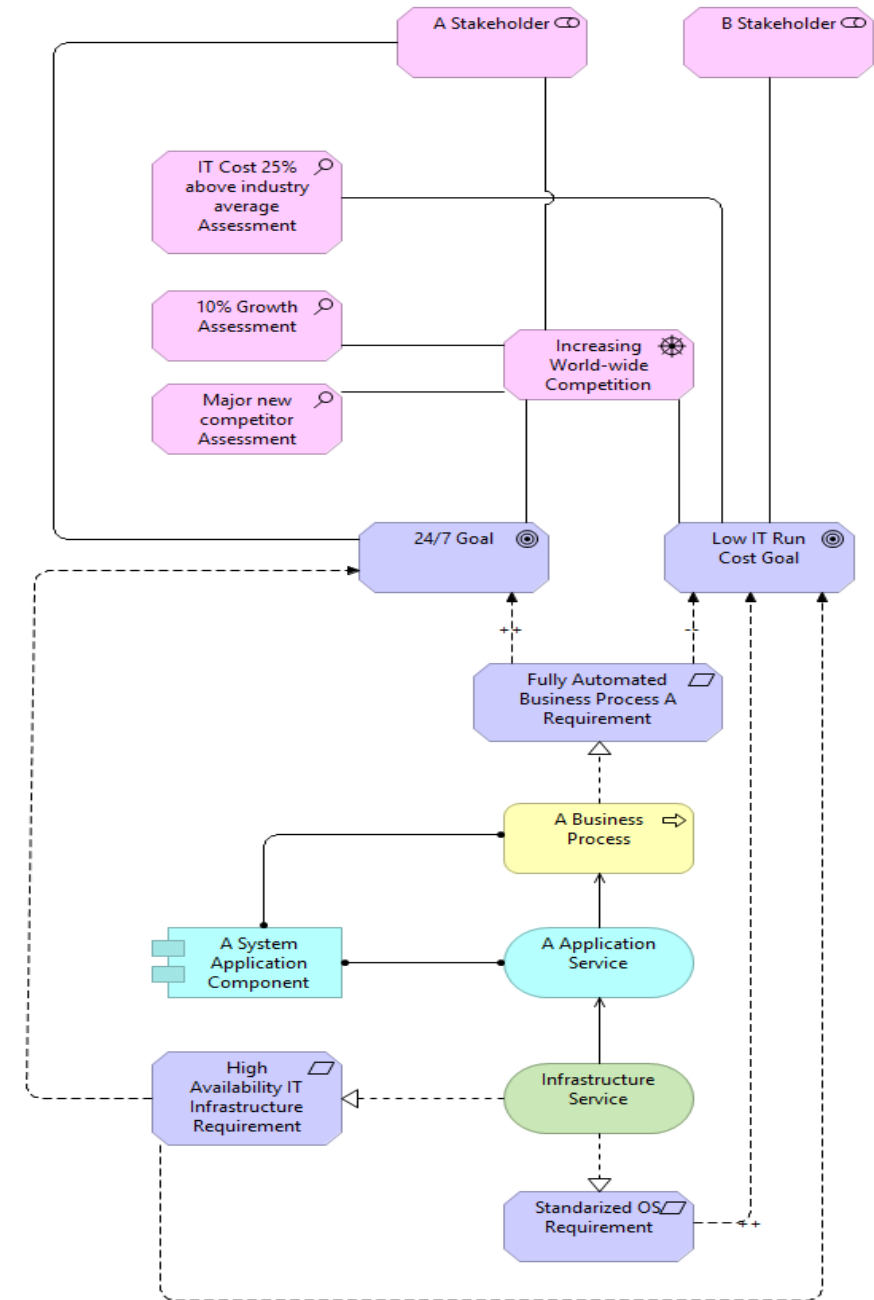
# Motivation

- Goal
  - End state that a stakeholder wants to achieve.
- Requirement
  - Obligatory aspect of what a system or process Realizes.
- Driver
  - Something that drive change in the organization.
  - May be external or internal.
- Assessment
  - Outcome of an analysis of a Driver
- Stakeholder
  - Role that is interested in achieving a Goal or might be associated with a Driver or an Assessment
- Constraint
  - A forbidden aspect of what a system or process Realizes.
- Principle
  - Sort-of Goal that is generalized Requirement



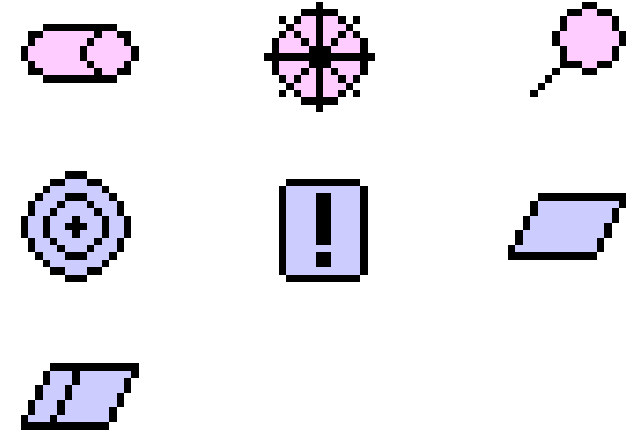
# Motivation

- Relations between elements are generally Associations
- Realization is used for:
  - Requirement, Constraint or Principle Realize a Goal
  - Requirement or Constraint Realize a Principle
  - Any core ArchiMate concept (except Value and Meaning) Realize a Requirement
- Influence relationship
  - To model the way Driver, Assessment, Goal, Principle, Requirement and Constraint can influence each other.
  - Normally, a label on the relationship is used to denote the type of influence.



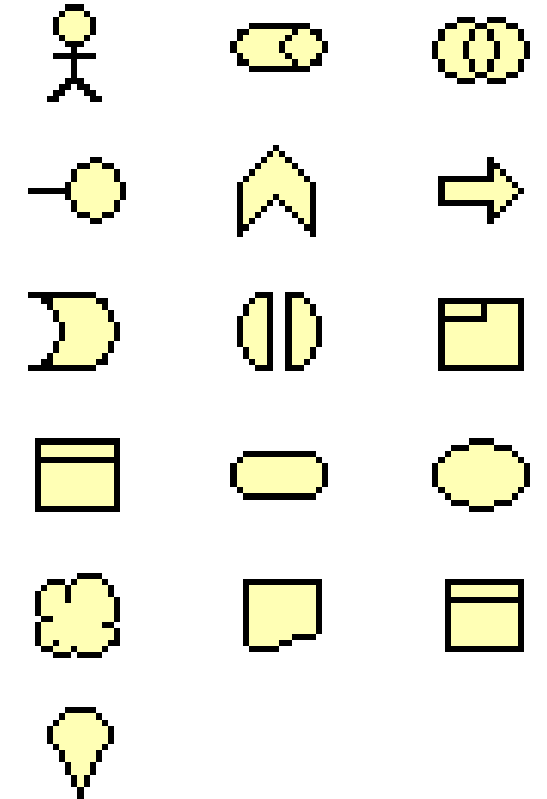
# Archi Requirements

- Stakeholder
- Driver
- Assessment
- Goal
- Principle
- Requirement
- Constraint



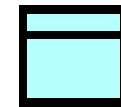
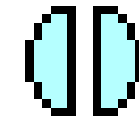
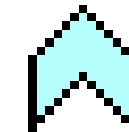
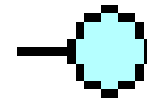
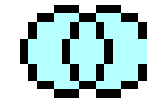
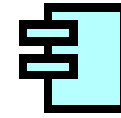
# Archi Business Elements

- Business Actor
- Business Role
- Business Collaboration
- Business Interface
- Business Function
- Business Process
- Business Event
- Business Interaction
- Product
- Contract
- Business Service
- Value
- Meaning
- Representation
- Business Object
- Location



# Archi Application Elements

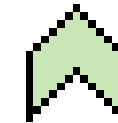
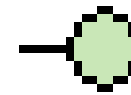
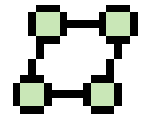
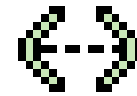
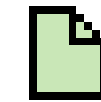
- Application Component
- Application Collaboration
- Application Interface
- Application Service
- Application Function
- Application Interaction
- Data Object





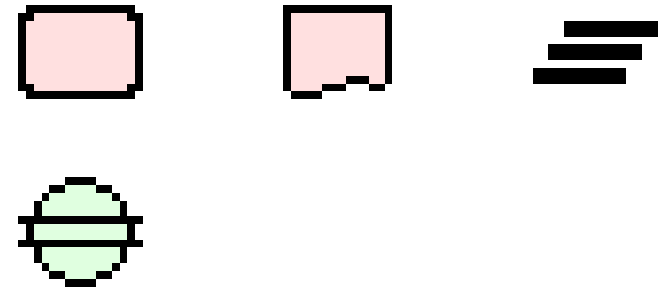
# Archi Infrastructure Elements

- Artifact
- Communication Path
- Network
- Infrastructure Interface
- Infrastructure Function
- Infrastructure Service
- Node
- System Software
- Device



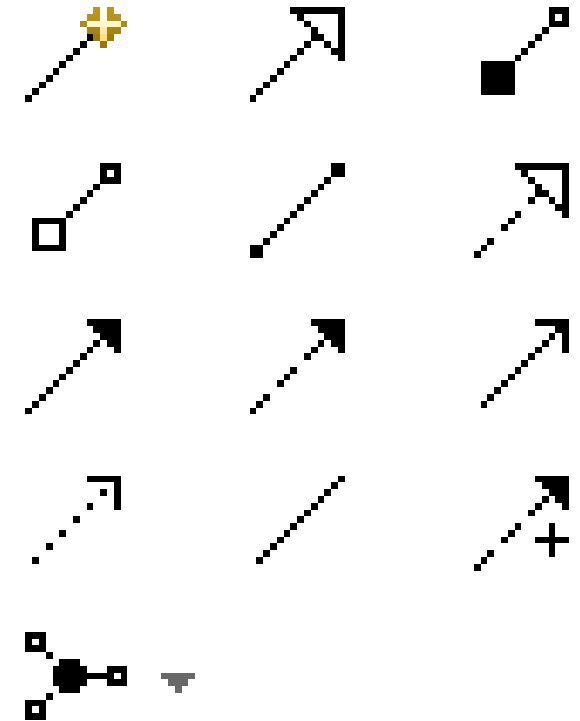
# Archi – DO WE NEED THIS??

- Work Package
- Deliverable
- Plateau
- Gap



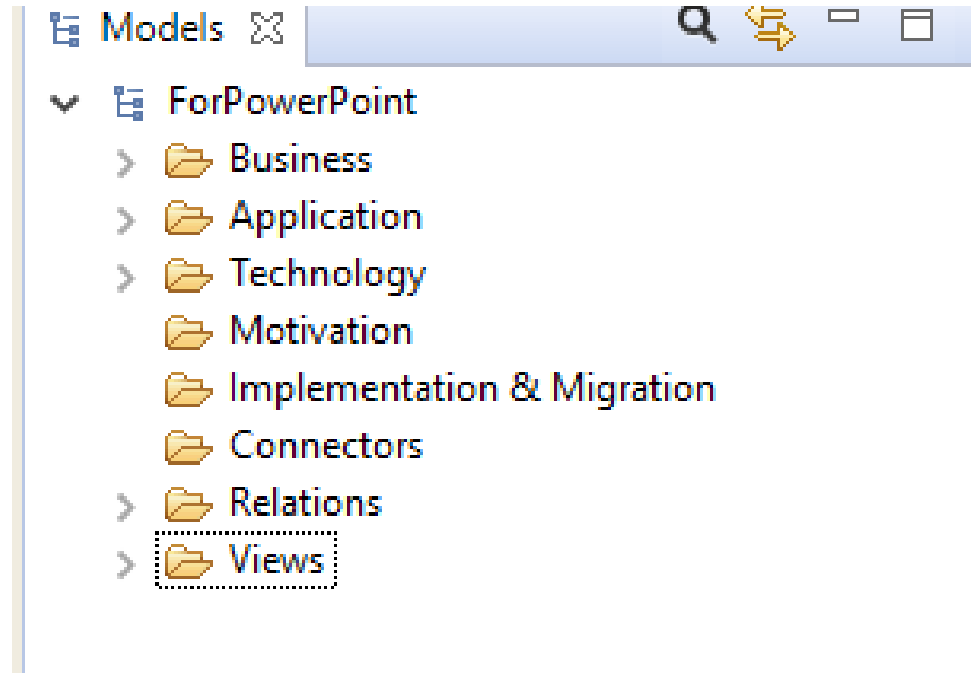
# Archi Connectors

- “Magic” Connector – Create connections automatically
- Specialization Relation
- Composition Relation
- Aggregation Relation
- Assignment Relation
- Realization Relation
- Triggering Relation
- Flow Relation
- Used By Relation
- Access Relation
- Association Relation
- Influence Relation
- Junction

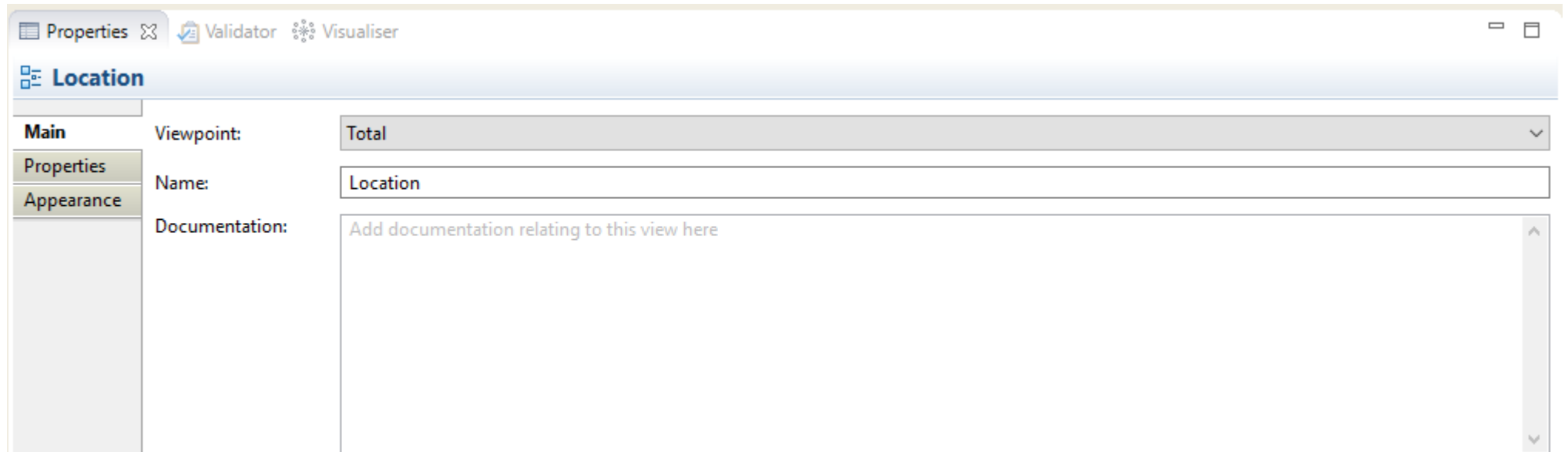


# Archi – Start of a new Model

- Views
  - Here is where you access your views and create new ones
  - Can also validate model
  - Can also create:
    - Sketch View
    - Folder
- Sketch View
  - Experimental feature of Archi
  - Meant to be the ArchiMate “Introductory viewpoint”
  - Don’t need to use this for this workshop



# Properties



- Selecting an element allows you to see and enter more detail properties for an element or connector

# Visualizer

