

# Architecting the Digital Enterprise

## *Making Standards Work*<sup>®</sup>

October 23, 2017

October 24, 2017

October 25, 2017

October 26, 2017

### TRACK: EA and Business Transformation Case Studies

4:00 PM - 4:45 PM

**Enterprise Modelling Using TOGAF, UML, IAF, ArchiMate, IT4IT, SparX for Business Transformation**

**Speakers:** [Alain Garsoux](#), [Alain De Preter](#)



Presentation of the [Labnaf Architecture Framework](#) at Open Group and its deployment at YPTO.

- Labnaf is a highly customizable architecture framework. The framework features some extensive and configurable modeling language and tools along with pragmatic on-line documentation.
- The strategy, architecture and planning standards that have been semantically merged into the Labnaf framework include [Archimate](#), [BPMN](#), [TOGAF](#), [SAFe](#), [Strategy Map](#), [BMM](#), [Gartner EA Stage Planning](#), [IT4IT](#), [PESTEL Analysis](#), [Porter's Five Forces Analysis](#) and [Business Model Generation](#).
- The Labnaf business transformation method, the modeling language and the repository structure are independent of their software implementation.
- The current software implementation is provided as a robust Sparx Enterprise Architect software extension. Sparx is the most common, yet affordable, modeling platform (more than 580 000 licenses worldwide).
- Open Group members were surprised to discover a large scale implementation of an architecture repository (at YPTO) delivering information towards the organisation.
- Open group member were exited that several standard frameworks and languages were integrated into one solution to support a business transformation.

## Case Study: Enterprise Modeling Using TOGAF, UML, IAF, ArchiMate and IT4IT to Support Business Transformation".

Alain De Preter - [alain.depreter@labnaf.one](mailto:alain.depreter@labnaf.one)

Alain Garsoux - [alain.garsoux@ypto.be](mailto:alain.garsoux@ypto.be)

Enterprise Architects @ Ypto,  
the IT company of the BE railways.



# Alain De Preter



## Career Summary

- YPTO (Railroad) - BE
- Brussels Airlines - BE
- Deutsche Bank – UK
- Deutsche Bank – BE
- Microsoft/Unisys Alliance – USA
- Unisys - BE
- Comedia – BE
- Ariane II - BE
- SWIFT - BE
- Borland – USA
- CPU2I - FR
- UCB - BE

## Expertise Summary

### Enabling

- Language Engineering
- Method Engineering
- Tool Engineering
- Modeling Coach

### Applying

- Enterprise & Solution Architecture
- Analysis, Design & Implementation

## Expectations

Demonstrate how a common language eases cross discipline collaboration

## Education/ Certification

- IT Engineer



# Alain Garsoux



## Career Summary

- Enterprise Architect @ NMBS/SNCB IT (Ypto)
- Enterprise Architect @ ING
- Solution Architect @ ING BE, Finance & Risk.
- Team Leader @ ING BE
- Portfolio Manager @ ING BE, Finance & Risk
- Business Analyst @ ING BE
- Project Leader @Carrefour NV
- Account Manager @ Cognos
- Inside Sales @ Merant
- Inside Sales, Marketing @ Progress Software.

## Expertise Summary

- Enterprise Architect.
  - Special attention to some value chains : IT, Facility Management, Safety, Finance & Risk, Procurement.
- People Coaching
- Business Intelligence
- Business Process

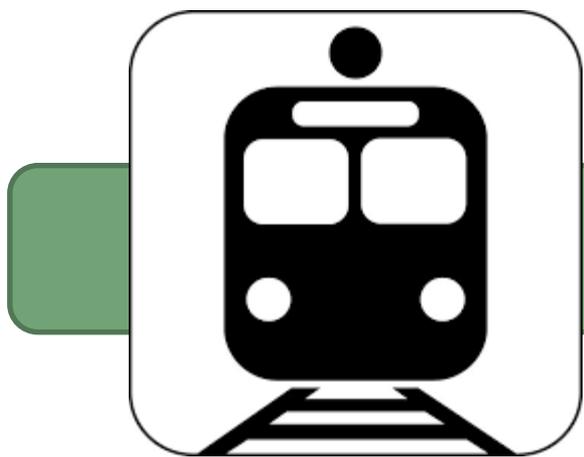
## Education/ Certification

- Commercial Ingeneer
- TOGAF 9 certified

## Expectations

Alain wants to share how architecture shapes an organization.





1

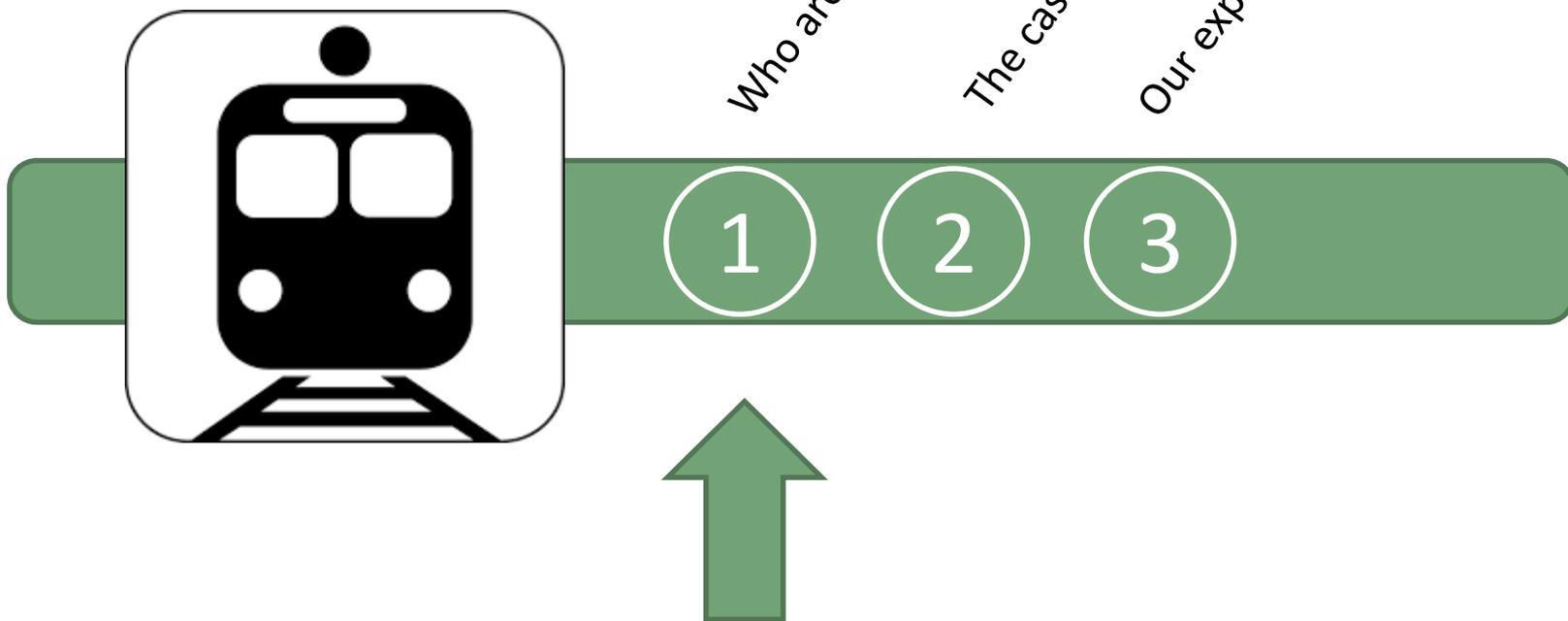
*Who are we?*

2

*The case*

3

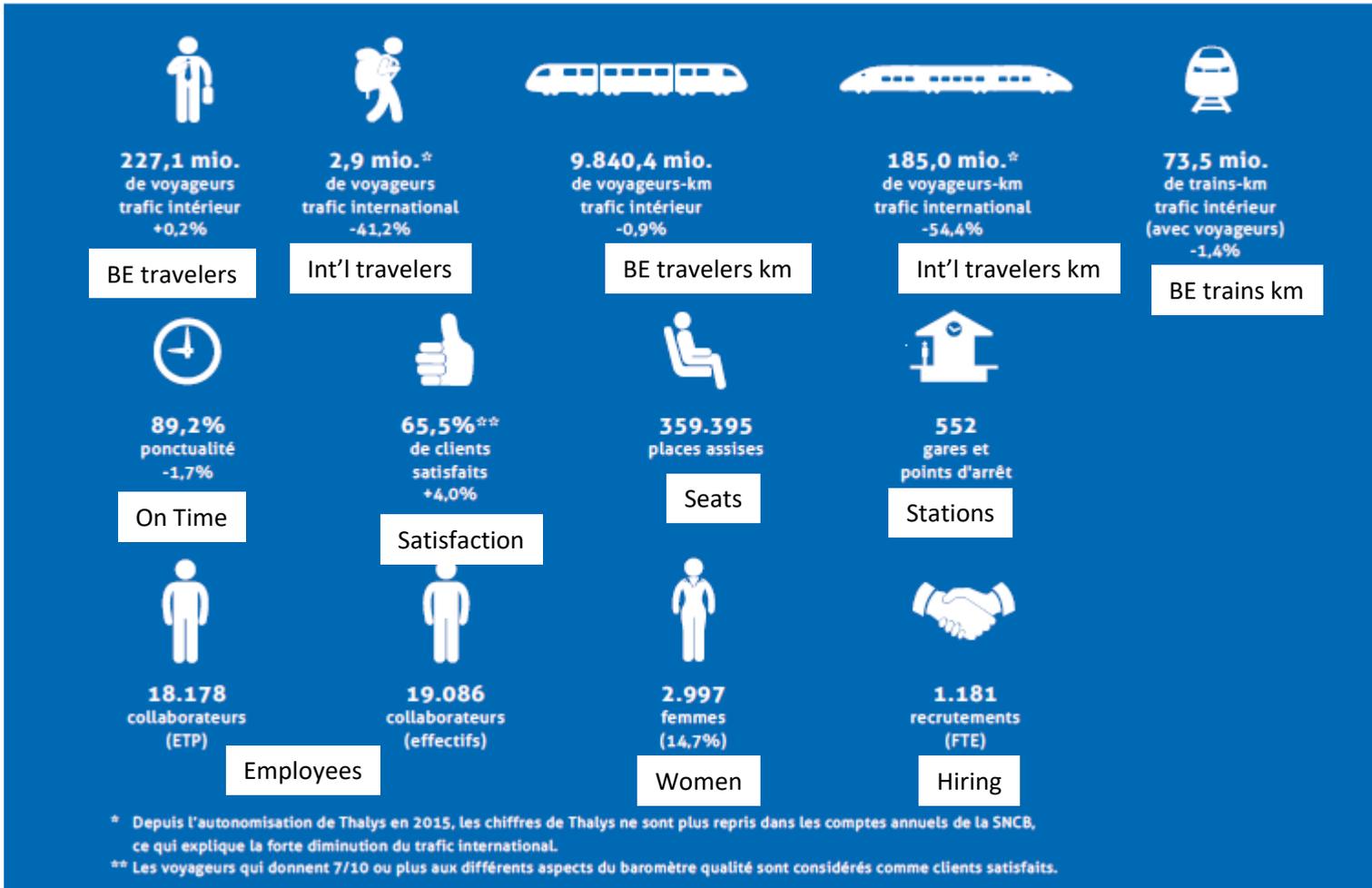
*Our experience*





# BE Railways

| 2016*         | Millions |
|---------------|----------|
| Income        | € 2 371  |
| Total balance | € 11 975 |
| Investments   | € 702    |



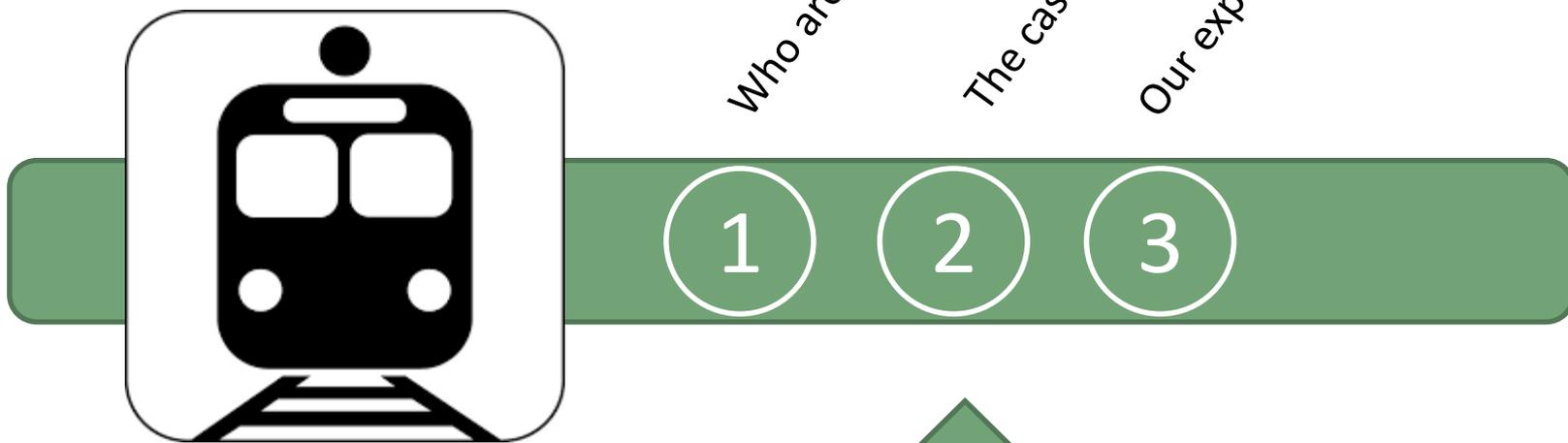
\* Source :  
annual report  
2016



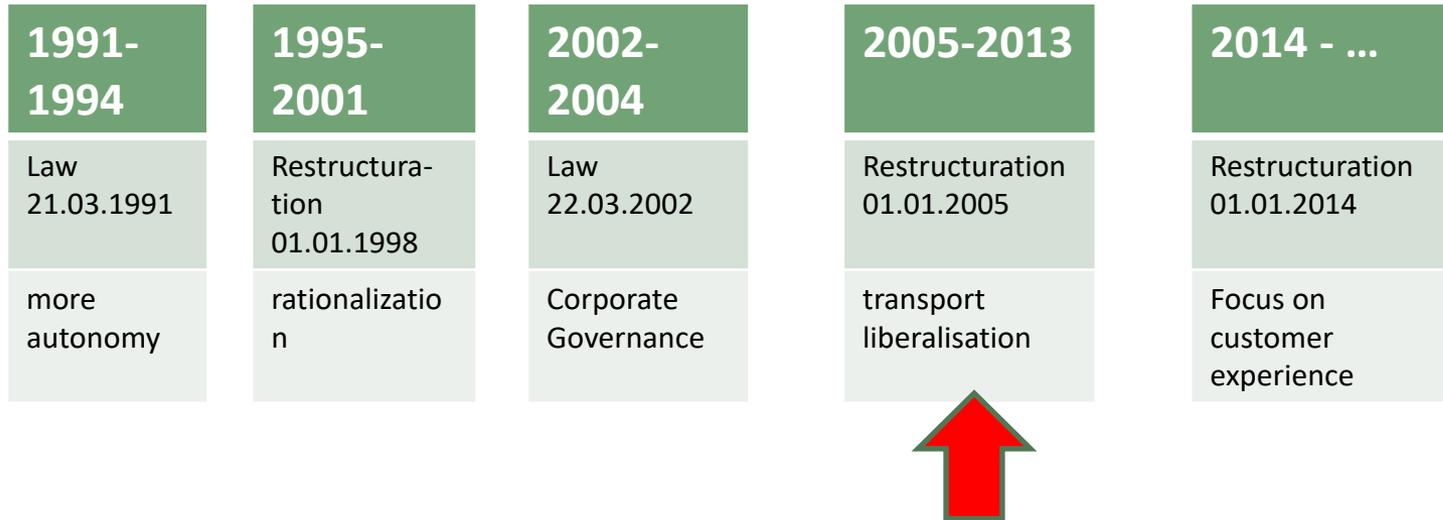
"Onze realisaties worden steeds zichtbaarder voor de treinreizigers. Dat daagt ons uit en maakt ons trots."

- Ypto is a subsidiary of BE railways.
- It delivers IT services.



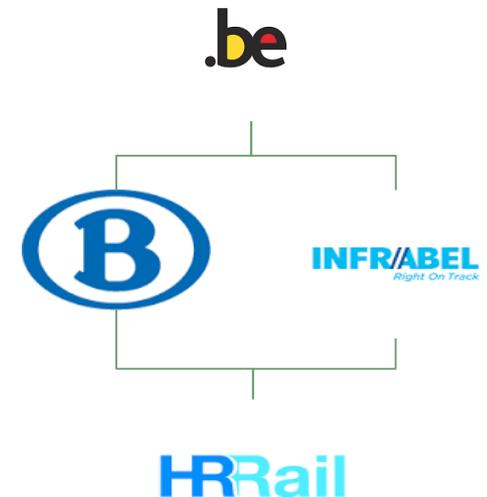


BE railways journey is impacted over years by BE government decisions and adoptions of European railways regulations. Our story started in 2005.



Following European regulations, BE railways had to split into 3 public companies.

| Companies | Business focus                    |
|-----------|-----------------------------------|
| INFRABEL  | Railroad infrastructure           |
| SNCB      | Rail operator (carry passengers)  |
| HR Rail   | HR services for SNCB and INFRABEL |



That also means splitting the IT organisation...

Our challenge is to create a new IT organization besides the fulfilment of usual business demand

- **Migrate thousands of shared INFRABEL-SNCB components** (applications, servers, network...) **to their own environments**
- **Problem:** Lack of visibility and traceability on the business and IT landscapes (lack of formal documentation)
- For example, there is very limited visibility on
  - What components exist
  - How they are structured
  - What they are used for
  - How they communicate and for what purpose
  - ...



Our challenge is to create a new IT organization besides the fulfilment of usual business demand

- Migrate thousands of shared components (applications in their own environment)

How to support this business transformation

**from an architecture view point?**

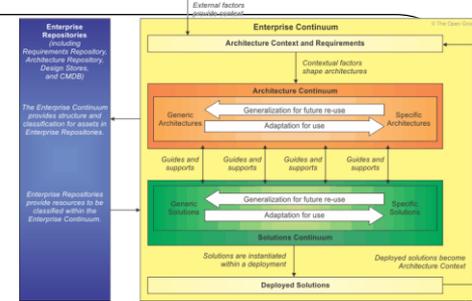
(visibility on the business formal documentation)

For example, there is very limited visibility on what components exist

- How they are structured
- What they are used for
- How they communicate and for what purpose
- ...



# YPTO (SNCB's IT) decided to create its Enterprise Continuum...



## Key aims

- Architecture Modeling + IT Portfolio Management
- Integrate architecture catalogs, attributes, descriptions...
- Modeling productivity and consistency
- Share enterprise visibility and traceability
- Ease planning and communication

## Decision to adopt

- Archimate, BPMN, UML, TOGAF, IT4IT & SAFe

UML was already used for analysis and design.



# ArchiMate, BPMN, UML, TOGAF, IT4IT, SAFe

## All are great and very useful but...

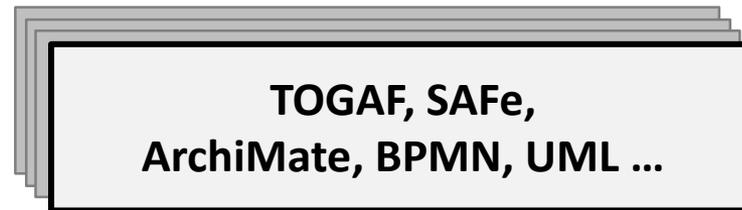
- Not integrated
- Not actionable as a whole (disconnected sets)
- Redundant semantics & terms (Data Object, Role, Process...)

## YPTO's conclusion

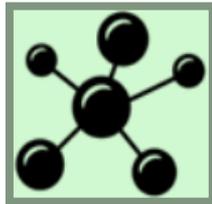
- Adopting each and every standard is not ideal
- Need a **single integrated company standard** to bring productivity, consistency, a single common language.

⇒ SNCB decided to go for a **single integrated framework** that is inspired by market standards and to **embed** it into the business of IT

# Building a framework inspired by market standards



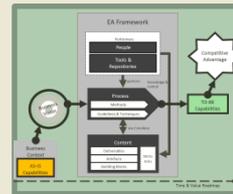
Filter, adapt, integrate,  
extend standards



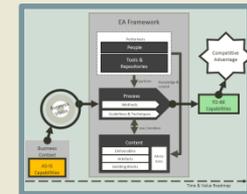
Normalize  
semantics



Reference Systems  
Semantics  
Automation By Nature



Tailor to  
customer needs

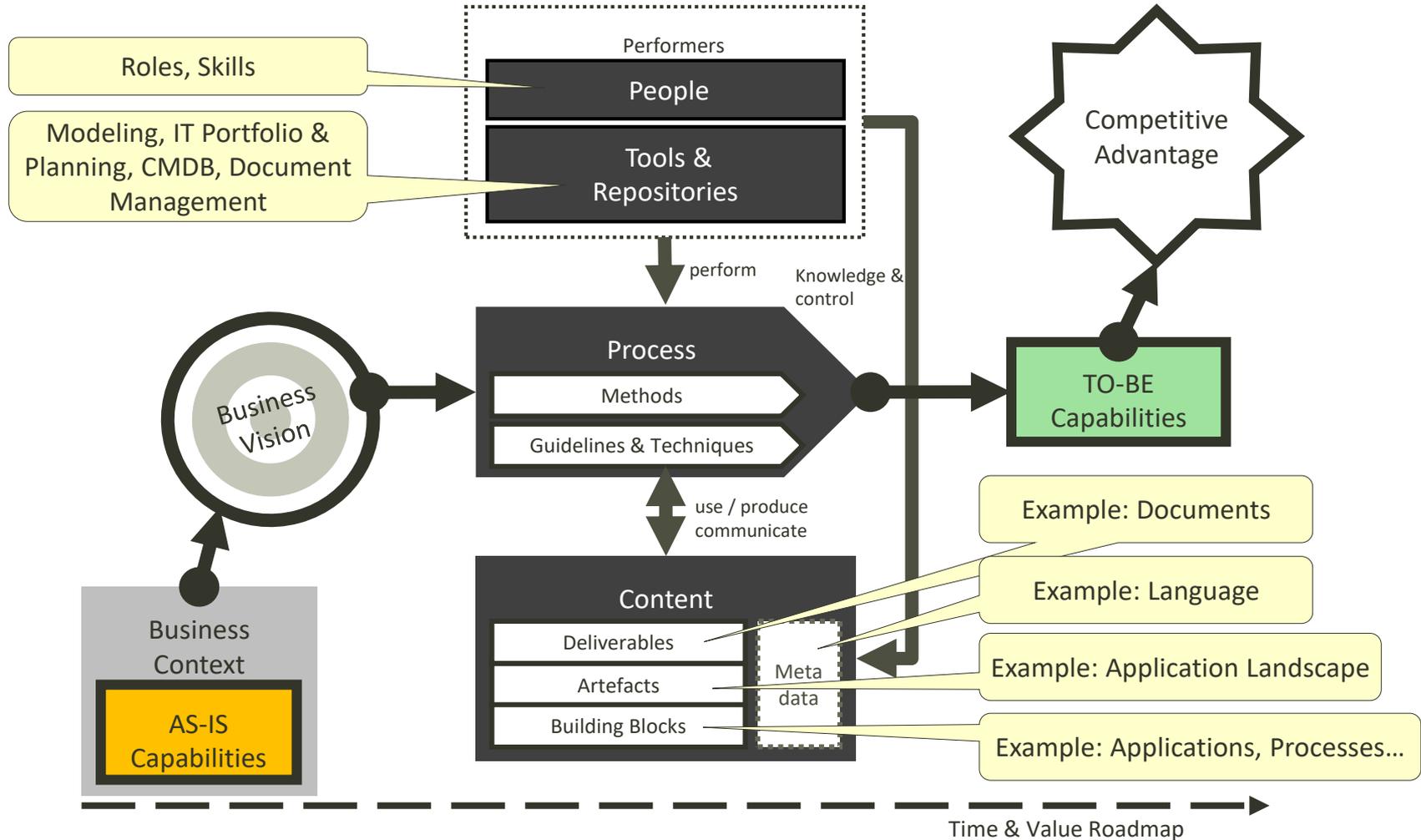


Reference Framework with  
Integrated Language

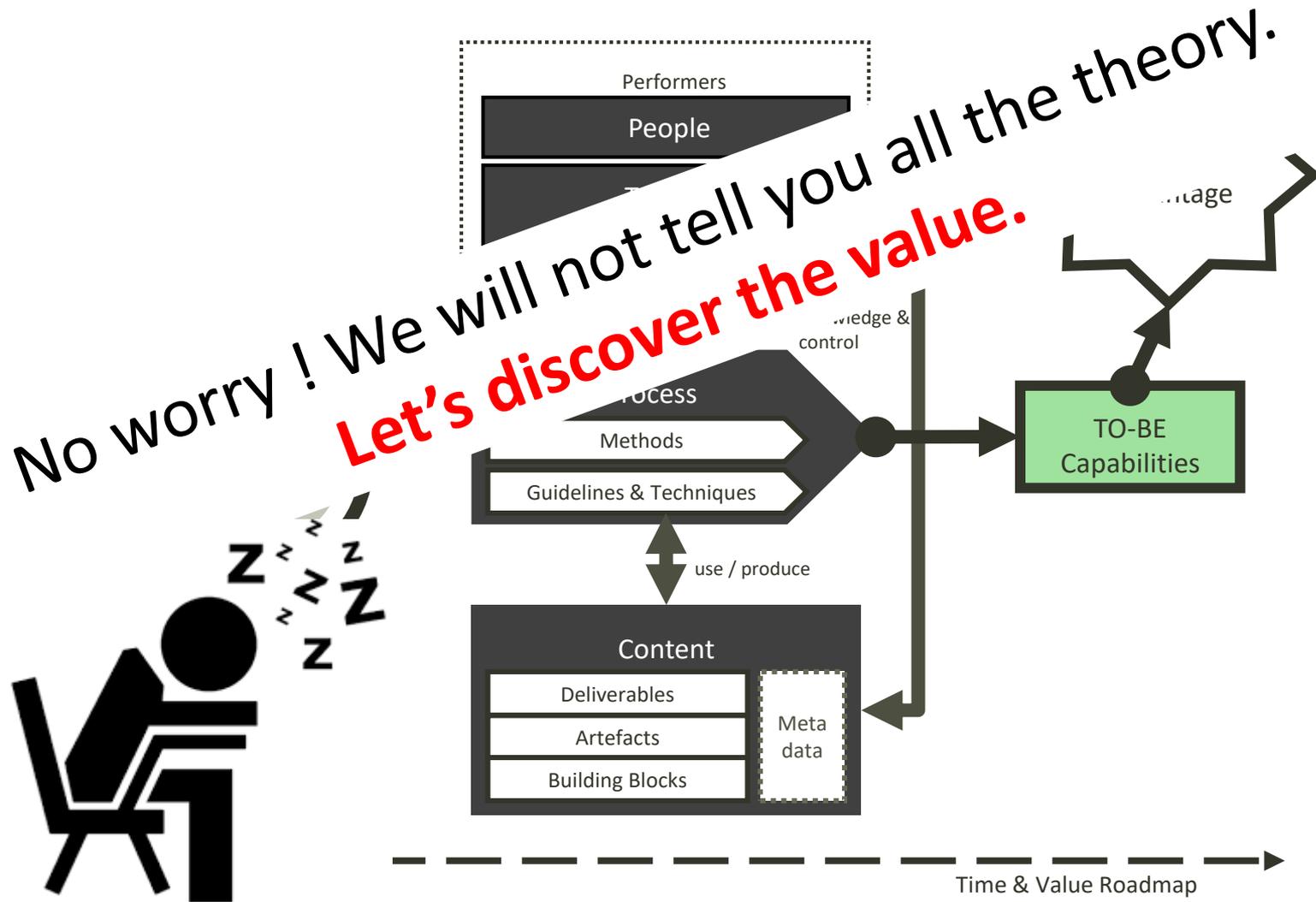
Architecture  
Framework being  
embedded

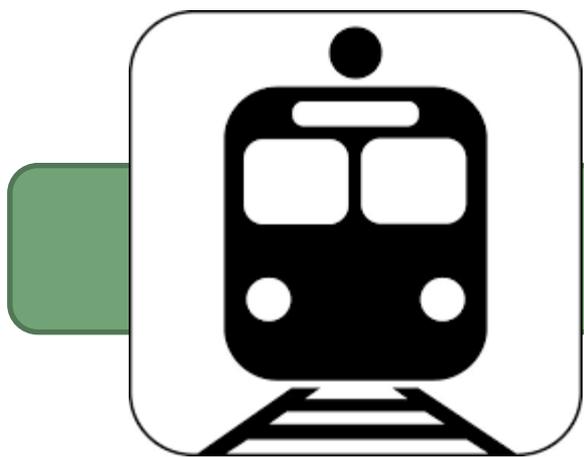


# What's in this framework?



# What's in this framework?





*Who are we?*

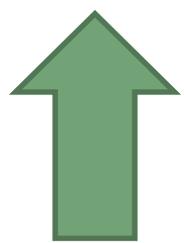
*The case*

*Our experience*

1

2

3



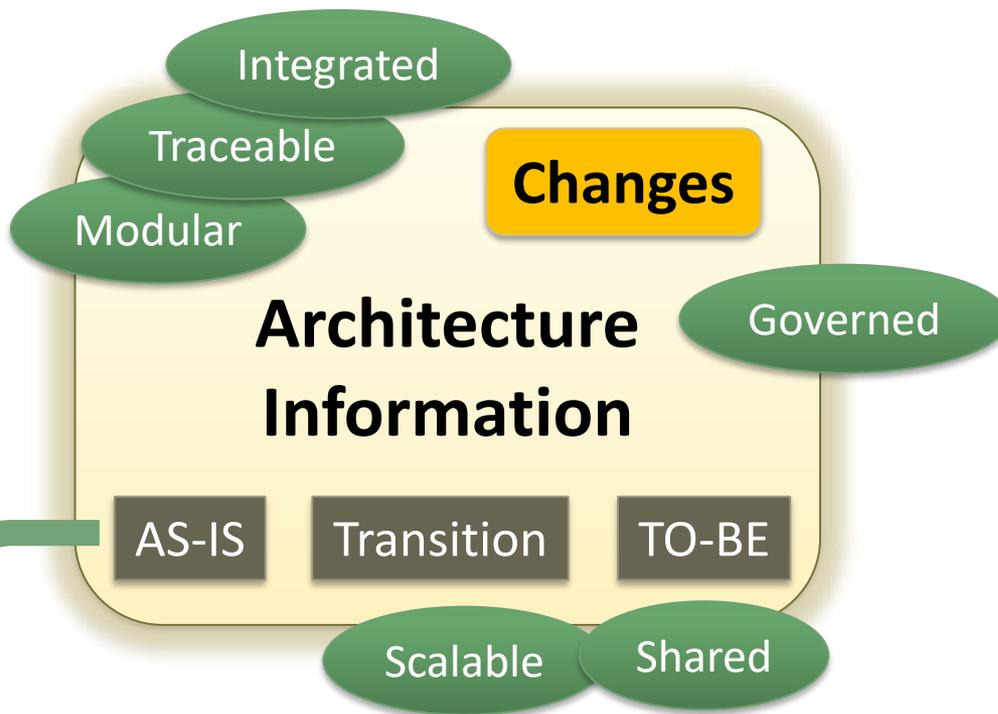
# Architecture views support change activities

- Define the model repository structure
- Define and organize the types of view (viewpoints)
- Clarify roles : Enterprise Architect / Solution Architect
- Integrate architecture views in project deliverables

This requires much communication and collaboration.

Integration in project deliverables is time consuming.  
It requires some management attention.

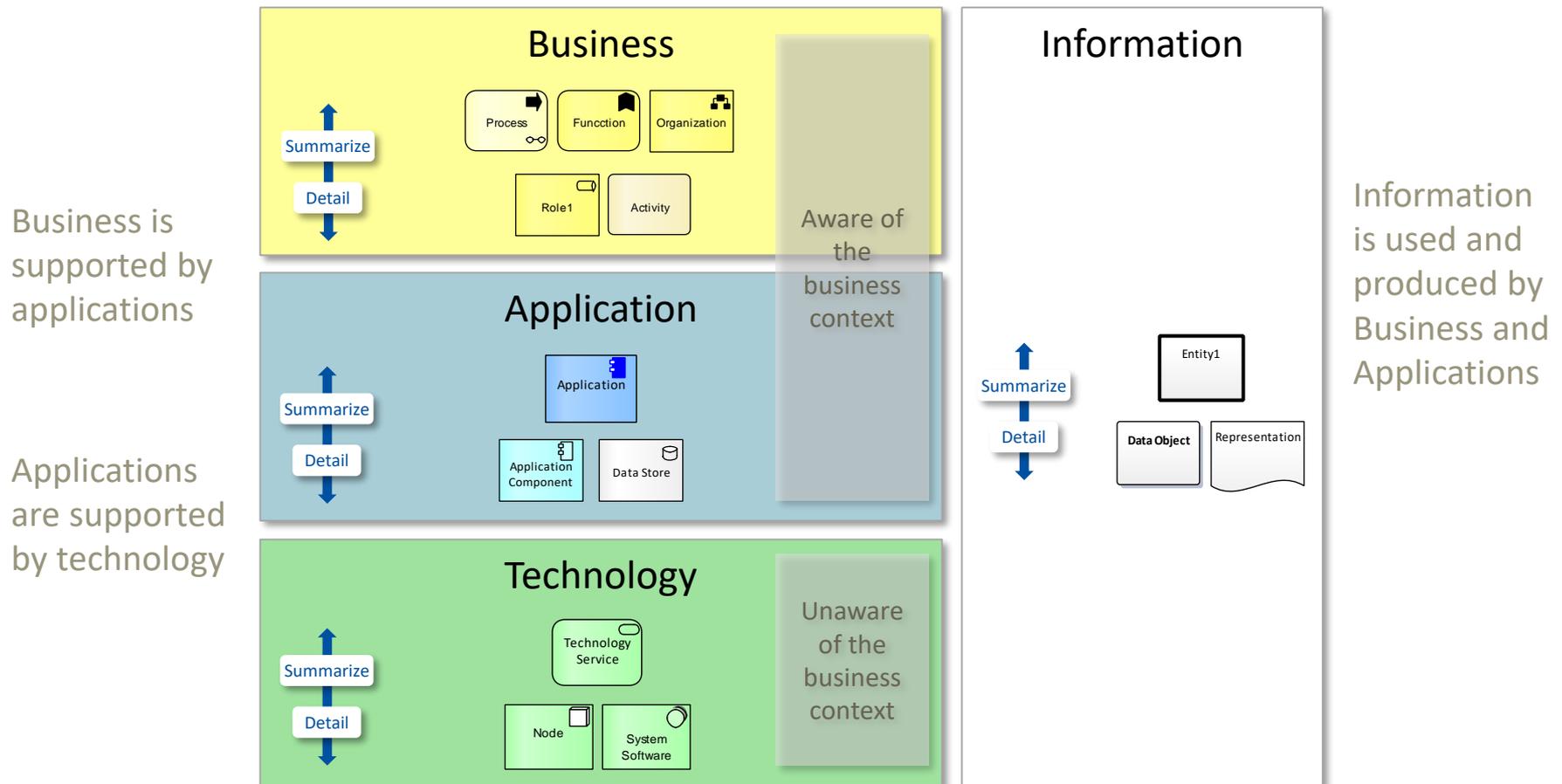
# Prescriptive Model Repository Structure



- ▷ ! Summary Views
- ▷ ! Vision
- ▷ ! Business
- ▷ ! Information
- ▷ ! Application
  - ▷ ! Application Functions
  - ▷ ! Applications
    - ▷ ! Application Platform1
      - ! APL - Application Platform1
      - ▷ ! Application Group1
        - ! AIA - Application Group1
        - ! APL - Application Group1
        - ▷ ! Application1
          - ! BOW - Application1
          - ! APL - Application1
          - ! AID - Application1
          - ! IST - Application1
          - ! APD - Application1
          - ! IND - Application1
          - ! Application Deployment Set1
          - ! Instance Deployment Set1
          - ! Application1 Database Node
          - ! Application1 Node
          - ! Application Component1
          - ! Application Component2
          - ! Data Store1
        - ! Application2
          - ! Shared Application Component2
          - ! Shared Data Store2
        - ! Application Group2
          - ! Shared Application Component1
          - ! Shared Data Store1
- ▷ ! Shared Data Stores
- ▷ ! Technology
- ▷ ! Analysis
- ! Design
- ! Projects

# Architecture Content is Structured Following Perspectives in order to Manage Complexity

An architecture perspective is similar to an IAF aspect area



# PROJECT-specific Viewpoints

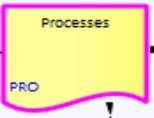
Vision

Click on any viewpoint to see its description

**Roles**

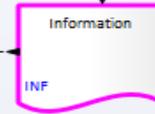
- Strategist
- Business Area Lead
- Subject Matter Expert
- Enterprise Architect
- Solution Architect

Business

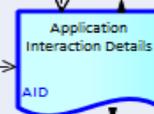
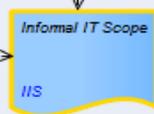


Viewpoints influence other viewpoints

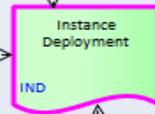
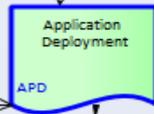
Information



Application



Technology



# Each type of architecture view uses specific types of element and connector

## Elements & Connectors

- Some come from standards
- Some were adapted
- Some are proprietary

**Examples**

**Activities**

- Activity
- Gateway
- Intermediate Event
- End Event
- Role
- Start Event
- Swimlane
- Data Object
- Representation
- Application Function
- Application Component
- Location

**Application Deployment**

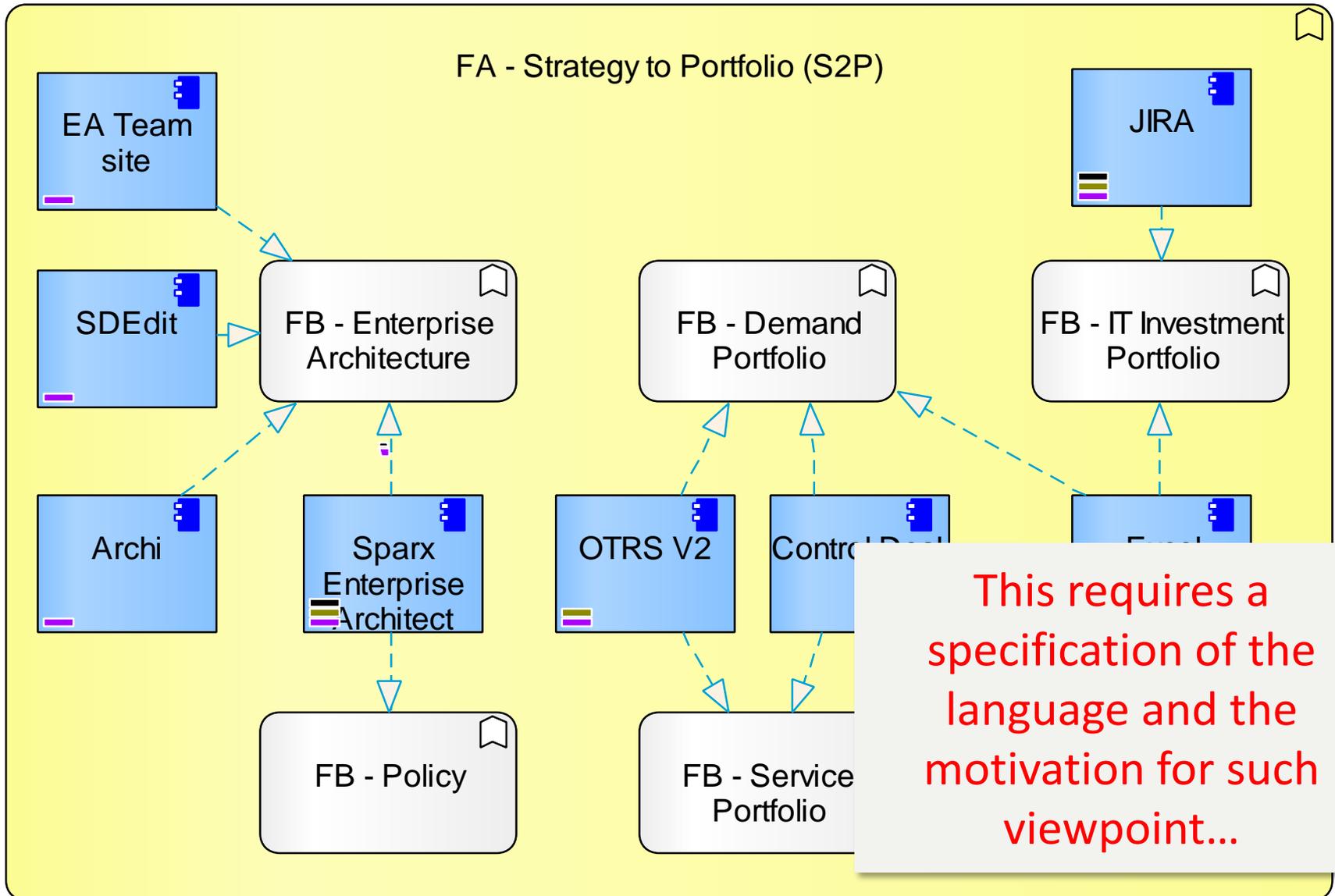
- Application Deployment Set
- Node
- System Software
- Application Component
- Location
- Data Store

**Application Deployment Connectors**

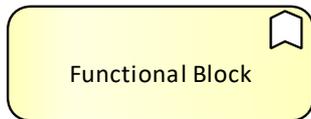
- Is deployed on
- Is part of
- Realizes
- Is dependent on
- Flow allowed by firewall

We use verbs instead of nouns to indicate the meaning of the connector's directions

# Sample “Functional Application Landscape” View



# Prescriptive language for “Functional Application Landscape” views



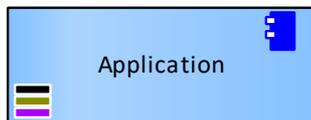
A Functional Block is a level 3 business function that belongs to some functional area.

The granularity and scope of a Functional Block is defined by identifying

- some homogenous set of information that the Functional Block is mastering
- a group of activities that fulfill the purpose of the functional block, that belong to some processes of same nature and that produce and use the information mastered by that Functional Block

*A business function is a behavior element that groups behavior based on a chosen set of criteria e.g. required business resources and/or skills, competencies, knowledge, etc.*

Inspired by Archimate



An application

- Is a self-contained unit of functionality as perceived by end-users
- Can be clearly mapped to some functional blocks
- Has its own specific set of application attribute values
- Is used by and billable to one or several Organizations
- Is owned by a single Organization
- Can be part of an Application Platform or an Application Group
- Encapsulates Applications Components and Application Interfaces
- Can exist at one or many specific points in time called "plateaus". Possible plateaus are AS-IS, TRANSITION and TO-BE.

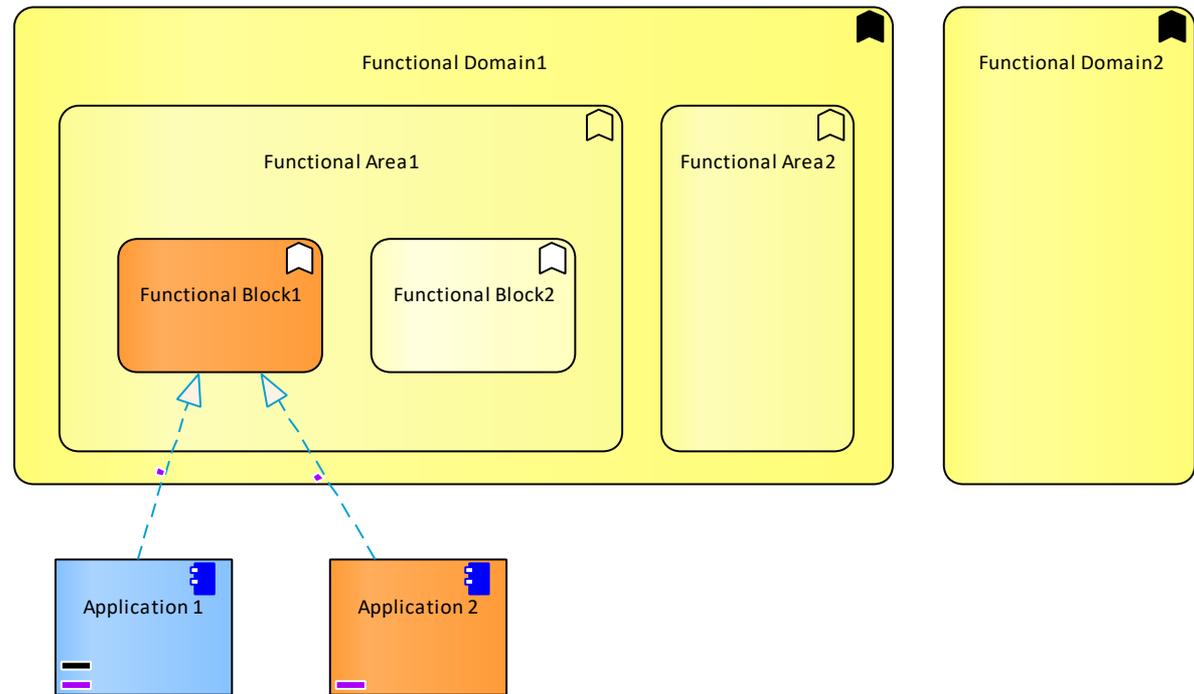
The name of an application component should preferably be a noun.



A **Realization** relationship indicates which concrete entities (“how”) realize which abstract entities (“what”). The realization relationship is used in a business operational sense (e.g., a role realizes a swim-lane of activities), but also in an IT context (e.g., an application realizes a functional block).

Inspired by UML & Archimate

# Motivation for creating “Functional Application Landscape” views



## This answers the following questions

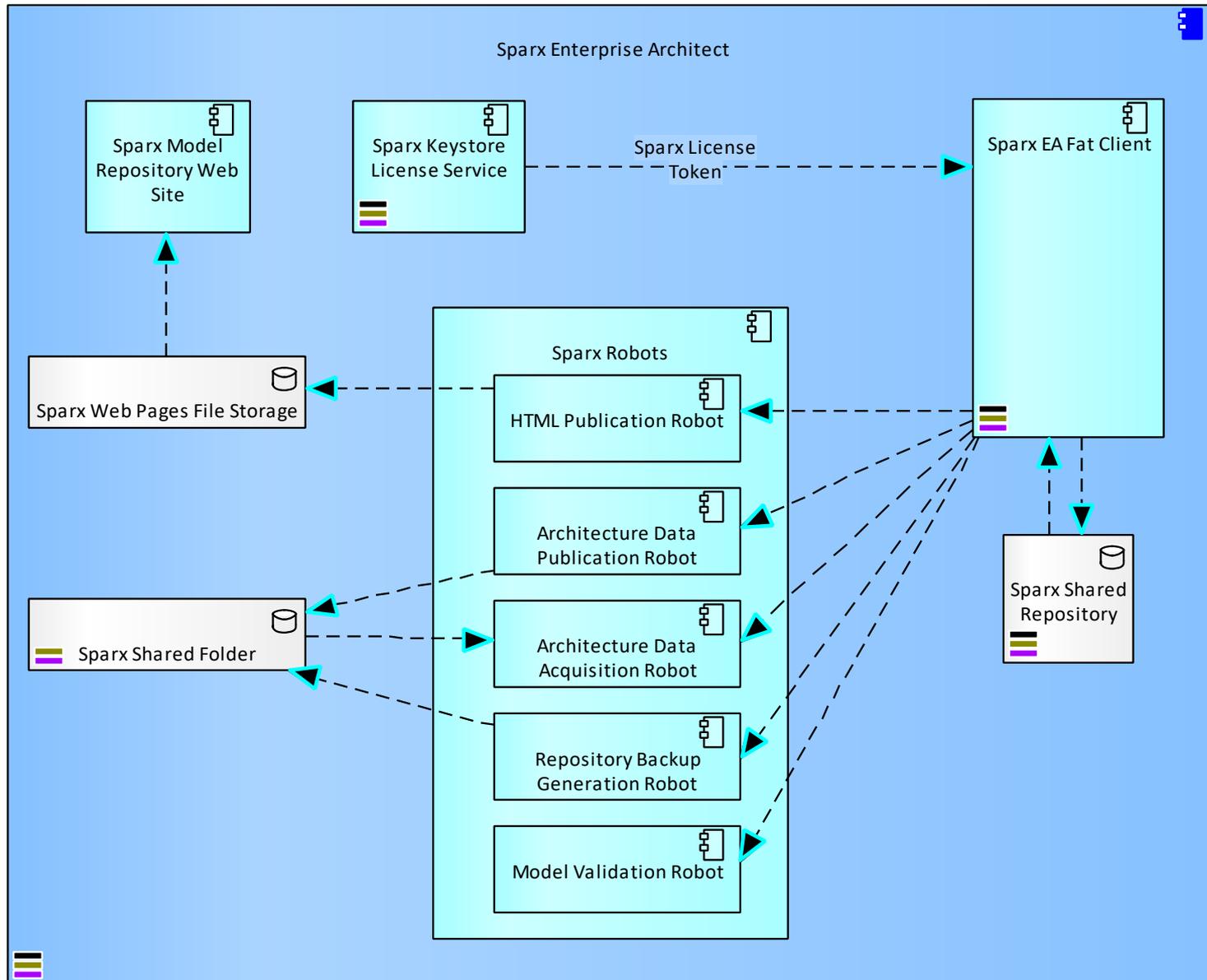
When used as enterprise-wide viewpoint

- Which **applications** support which **functional blocks**?

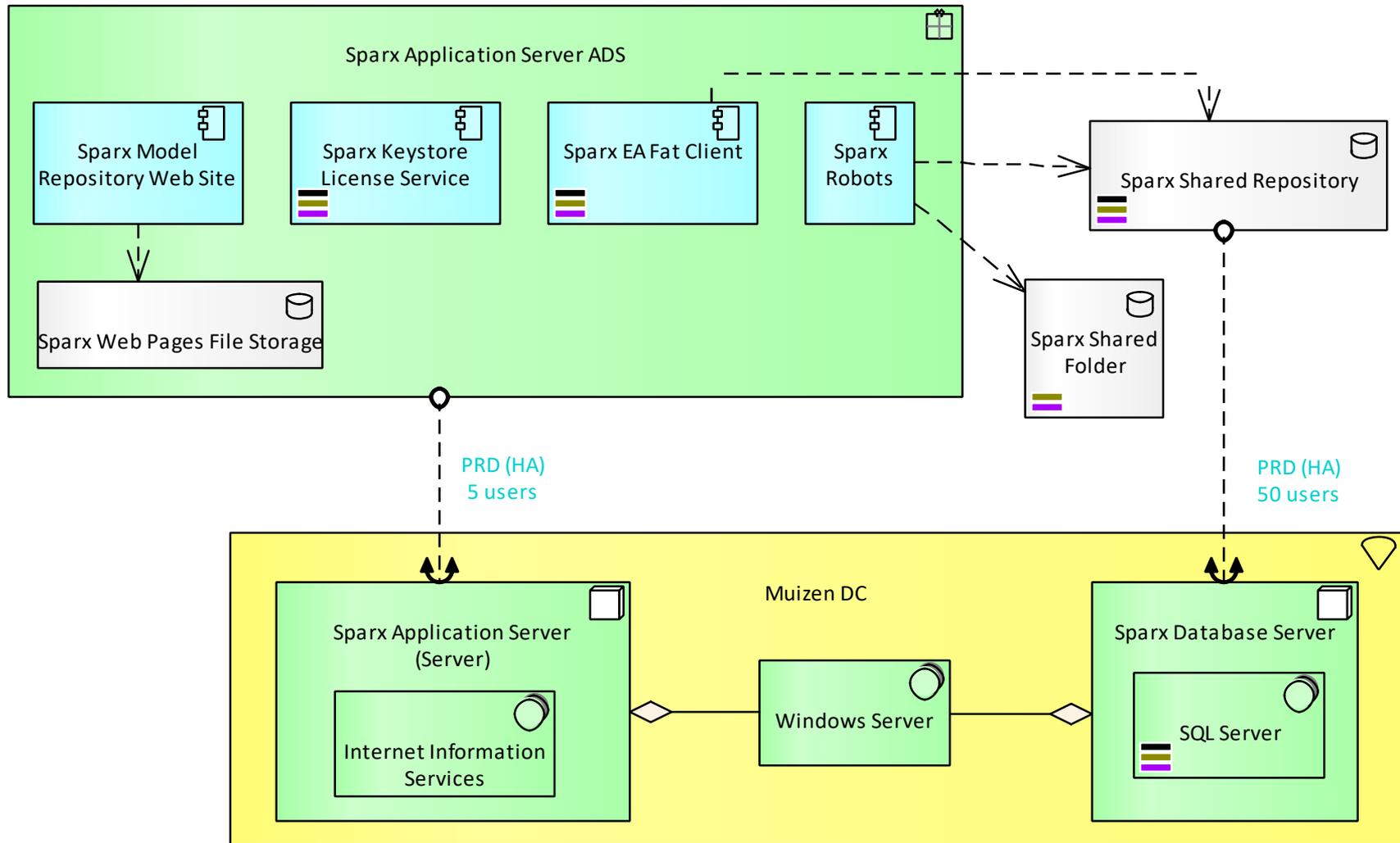
When used as project-specific viewpoint

- Which **applications** are/will automate the **functional blocks** inside the scope of this project?

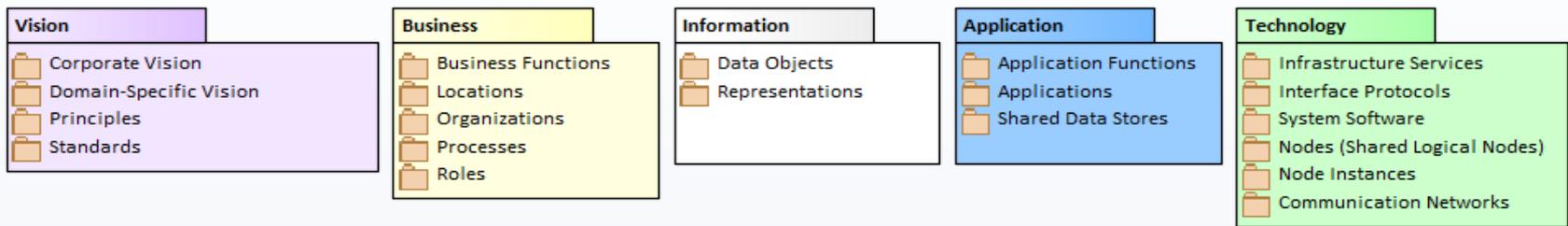
# Sample “Application Interaction Details” View



# Sample "Application Deployment" View



## Catalogs



## Enterprise-wide Viewpoints

| Vision                          | Business                        | Information            | Application                      | Technology                   |
|---------------------------------|---------------------------------|------------------------|----------------------------------|------------------------------|
| Viewpoints                      | Viewpoints                      | Viewpoints             | Viewpoints                       | Viewpoints                   |
| Context                         | Processes                       | Entities               | Application Landscape            | Standard Technology Services |
| Corporate Strategic Foundations | Functional Landscape            | Information            | Functional Application Landscape | Application Deployment       |
| Corporate Strategy Map          | Functional Interactions         | Information Processors | Application Interactions         | Connectivity                 |
| Corporate Goals                 | Business Organization Structure | Information Storage    | Application Interaction Details  | Instance Deployment          |
| Domain-specific Goals           | Business Ownership              |                        |                                  |                              |
| Domain-specific Demands         |                                 |                        |                                  |                              |
| Requirements & Planning         |                                 |                        |                                  |                              |
| Principles                      |                                 |                        |                                  |                              |
| Standards                       |                                 |                        |                                  |                              |
| Showing 1 - 9 of 9 items        | Showing 1 - 5 of 5 items        |                        |                                  |                              |

[Architecture Management Viewpoint](#)

+ Standard HTML, Excel & Word reports published daily on intranet

## Project-specific Viewpoints

| Vision                   | Business                 | Information              | Application                      | Technology               |
|--------------------------|--------------------------|--------------------------|----------------------------------|--------------------------|
| Viewpoints               | Viewpoints               | Viewpoints               | Viewpoints                       | Viewpoints               |
| Requirements & Planning  | Informal Business Scope  | Information              | Informal IT Scope                | Application Deployment   |
|                          | Functional Landscape     | Information Storage      | Application Landscape            | Connectivity             |
|                          | Processes                |                          | Functional Application Landscape | Instance Deployment      |
|                          | Functional Interactions  |                          | Application Interactions         |                          |
|                          | Activities               |                          | Application Functions            |                          |
|                          |                          |                          | Application Interaction Details  |                          |
| Showing 1 - 1 of 1 items | Showing 1 - 5 of 5 items | Showing 1 - 2 of 2 items | Showing 1 - 6 of 6 items         | Showing 1 - 3 of 3 items |



For further information about  
the language...

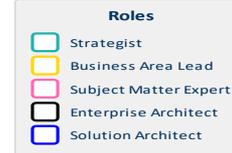
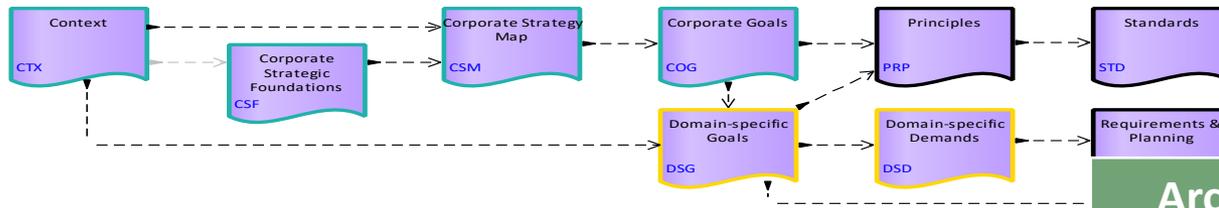
**Detailed documentation and contact  
information are available here:**

**[www.Labnaf.one](http://www.Labnaf.one)**

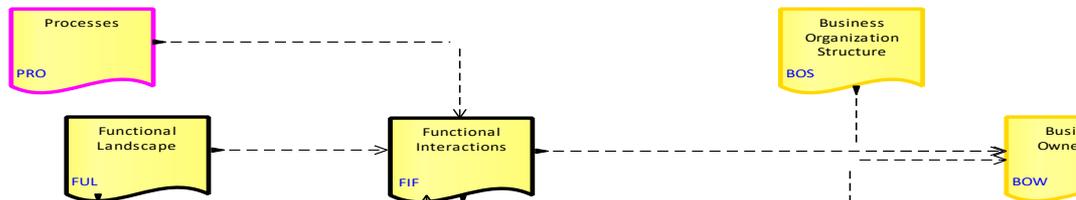
# BACKUP SLIDES

Vision

Click on any viewpoint to see its description



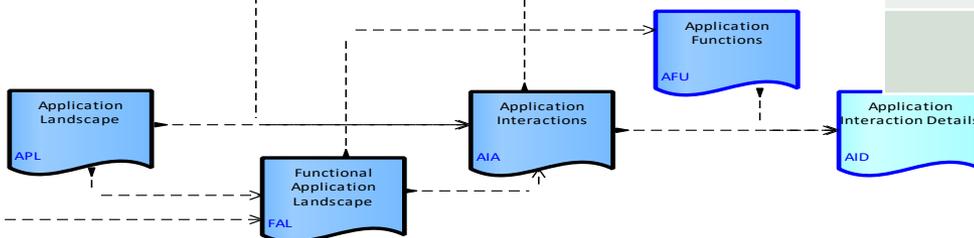
Business



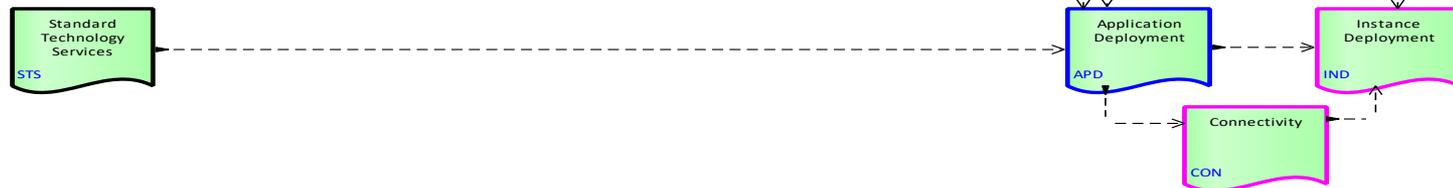
Information



Application



Technology



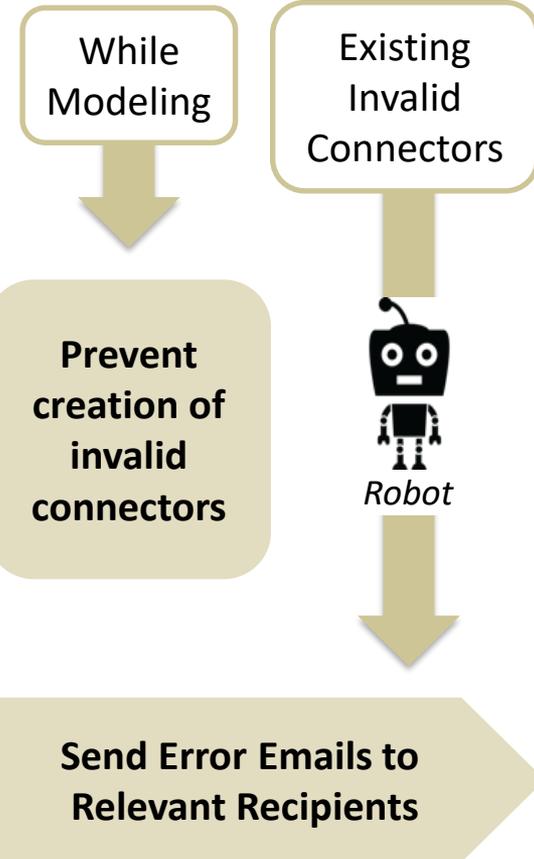
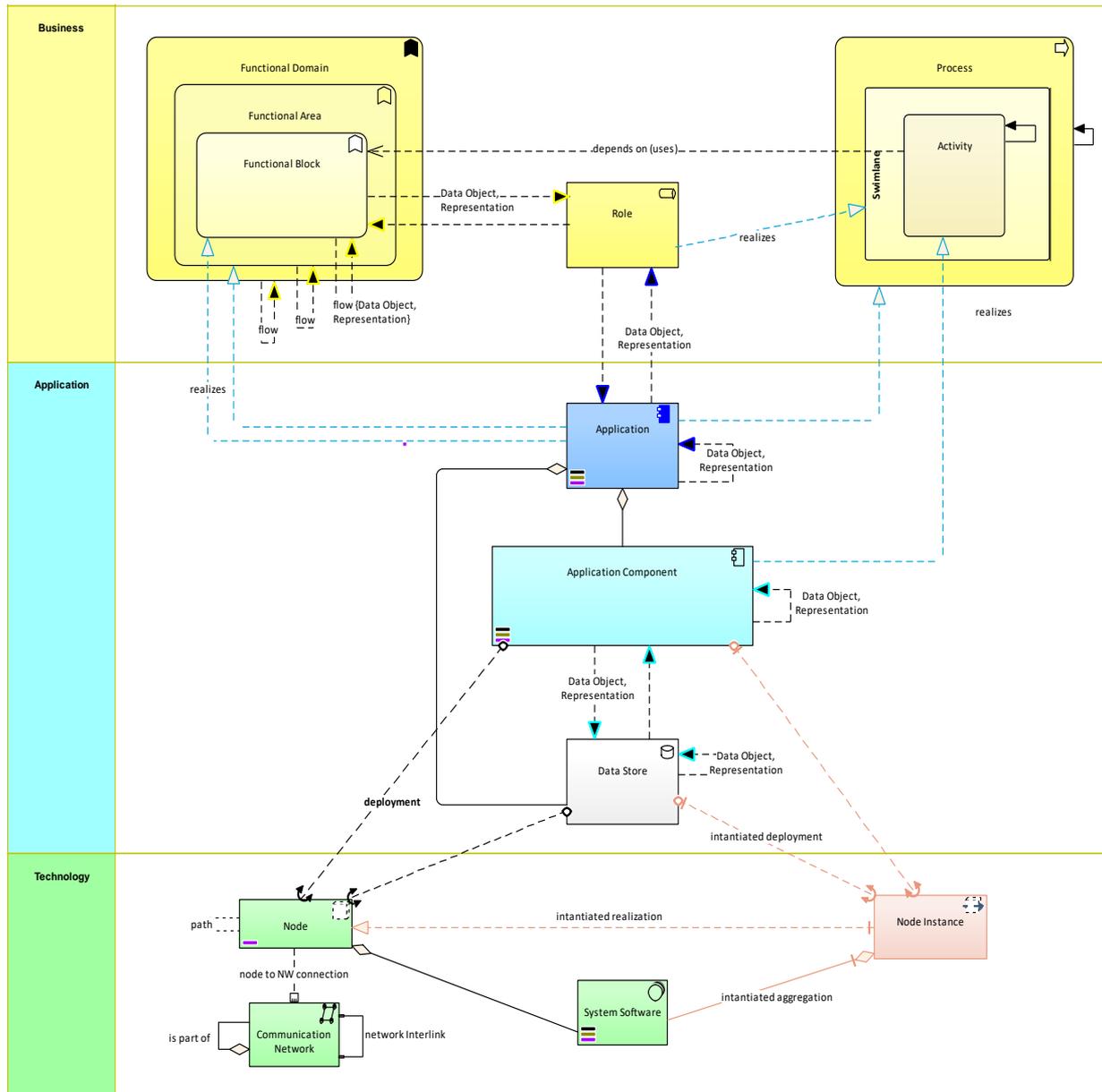
Architecture Management



**Architecture viewpoints organized following architecture perspectives**

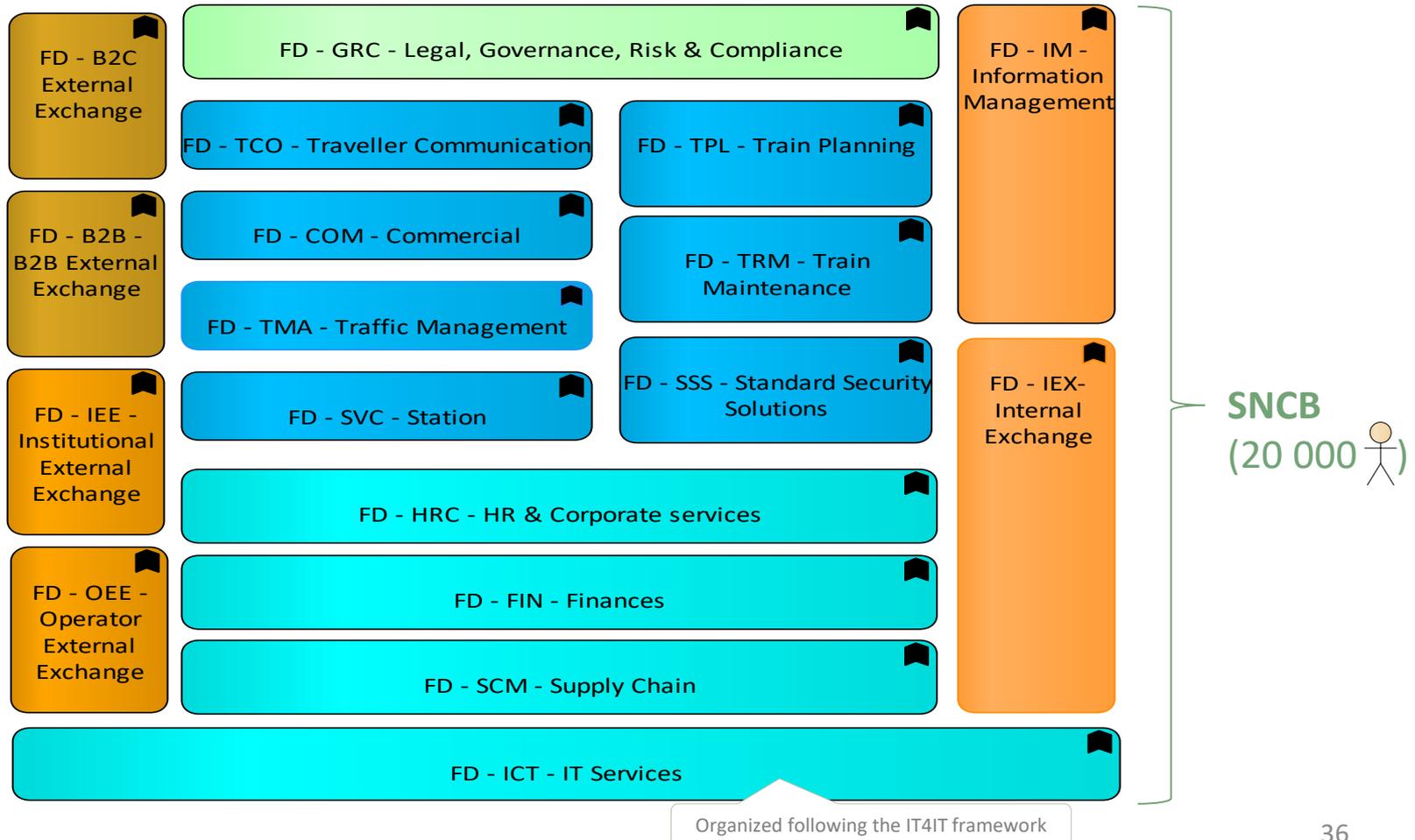
- Vision
- Business
- Information
- Application
- Technology

# Formalized Relationships



# Background: SNCB & YPTO

- Around 2009, SNCB created its own IT company: YPTO



# Merging Standards & Best Practices

1. **Use a generic systems ontology as a reference for core semantics**
2. **Use standards as references for frameworks and languages** (TOGAF, SAFe Archimate, BPMN, UML...)
3. **Define the strategy and architecture process**
4. **Identify the related viewpoints needed to execute the process**
5. **Define a metamodel of concepts needed throughout the process**
6. **Identify language semantics needed to illustrate each viewpoints**
7. **Select matching semantics, terms and language items from standards**
8. **Avoid redundancies across standard languages** e.g. Role, Data Object, Process...
9. **Add missing language constructs** (standard, organization, application, data store, deployment, virtualization, technology as a service...)
10. **Merge the resulting language constructs into a modeling language and a language metamodel**
11. **Organize viewpoints into flows and hierarchies** (levels of detail)
12. **Define the model repository structure** to store model elements and views
13. **Develop, experiment and document iteratively** in the shared repository and tool

# Views and viewpoints – Textual definitions

Definitions based on [Wikipedia](#)

In the [engineering](#) of physically intensive systems, viewpoints often correspond to capabilities and responsibilities within the engineering organization.

Most complex system specifications are so extensive that no single individual can fully comprehend all aspects of the specifications. Furthermore, we all have different interests in a given system and different reasons for examining the [system's specifications](#).

- [Viewpoint](#) is a systems engineering concept that **describes a partitioning of concerns in system restricted to a particular set of concerns**. Adoption of a viewpoint is usable so that issues in those aspects can be addressed separately. A good selection of viewpoints also partitions the design of the system into specific areas of Expertise. Viewpoints provide the conventions, rules, and languages for constructing, presenting and analysing views.
- A [view](#) of a system is **a representation of the system from the perspective of a viewpoint**. This viewpoint on a system involves a perspective focusing on specific concerns regarding the system, which suppresses details to provide a simplified model having only those elements related to the concerns of the viewpoint. It is the sum of all views together that describes a system sufficiently. The view uses the conventions, rules and language defined by the viewpoint to document the relevant aspect of the system.

