Seven ways to improve business development with Sparx EA

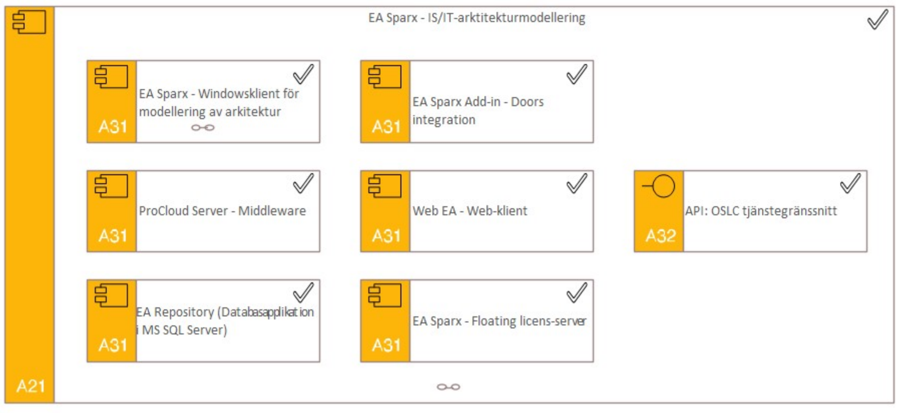
Per Carlsson

***Enterprise Architect** - SKB, the Swedish Nuclear Fuel and **Waste Management** Co*

Building engineer (1992), IASA IT-architect (2012)
+25 years in regulated industry – engineering and IT
Energy/Nuclear (16 years) Life science (10 years)

- Strategy and target architecture
- Data center and distributed network architecture
- Solution architecture System design
- Architecture framework and description
- Continuous integration and Continuous development
- Content management
- Author (IASA), Hidden structure analysis (KTH Stockholm, Université de Paris)

Seven ways to improve architecture development with Sparx EA



EA Sparx - IS/IT-arktitekturmodellering ✓

A21

A31 EA Sparx - Windowsklient för modellering av arkitektur ✓

A31 EA Sparx Add-in - Doors integration ✓

A31 ProCloud Server - Middleware ✓

A31 Web EA - Web-klient ✓

A32 API: OSLC tjänstegränssnitt ✓

A31 EA Repository (Databasapplikat ion i MS SQL Server) ✓

A31 EA Sparx - Floating licens-server ✓

Per Carlsson

***Enterprise Architect** - SKB, the Swedish Nuclear Fuel and **Waste Management** Co*

Building engineer (1992), IASA IT-architect (2012)
+25 years in regulated industry – engineering and IT
Energy/Nuclear (16 years) Life science (10 years)

- Strategy and target architecture
- Data center and distributed network architecture
- Solution architecture System design
- Architecture framework and description
- Continuous integration and Continuous development
- Content management
- Author (IASA), Hidden structure analysis (KTH Stockholm, Université de Paris)

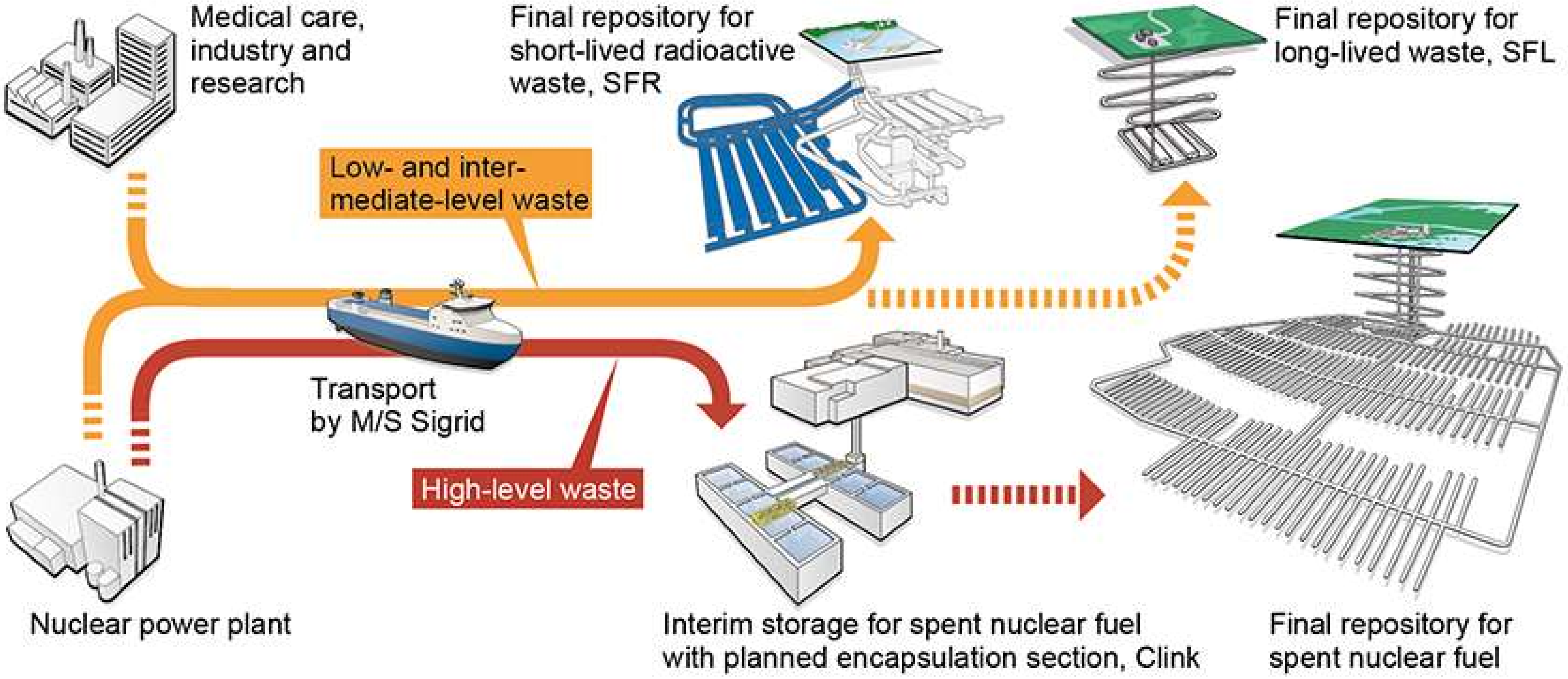
…We love;
- Lists
- Matrices
- Models

Get ready for a set of slides!

https://se.linkedin.com/in/percarlsson

The Enterprise

SKB, the Swedish Nuclear Fuel and Waste Management Co

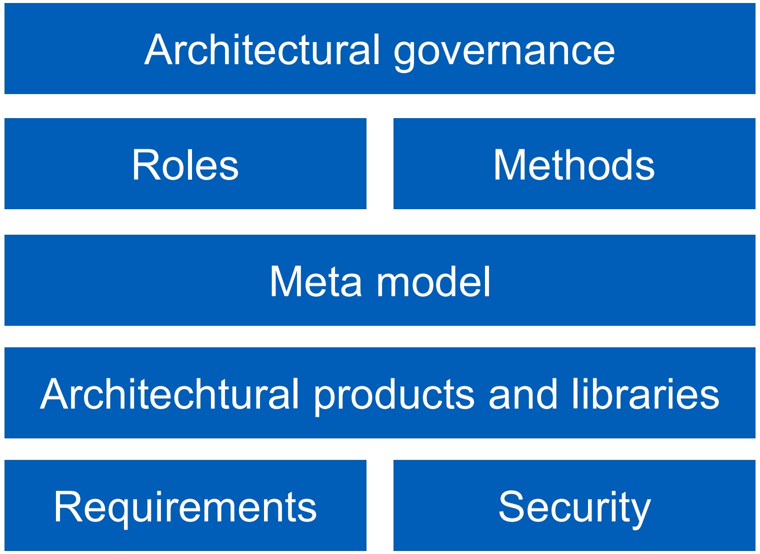

Medical care, industry and research
Final repository for short-lived radioactive waste, SFR
Final repository for long-lived waste, SFL
Low- and intermediate-level waste
Transport by M/S Sigrid
High-level waste
Nuclear power plant
Interim storage for spent nuclear fuel with planned encapsulation section, Clink
Final repository for spent nuclear fuel

…a quite cool capability flow revealing the enterprise "as is" and "to be"!

Pending of what you think is cool…

Presentation at-a-glance

Overview – Seven ways to improve business development with Sparx EA



Architectural governance

Roles

Methods

Meta model

Architechtural products and libraries

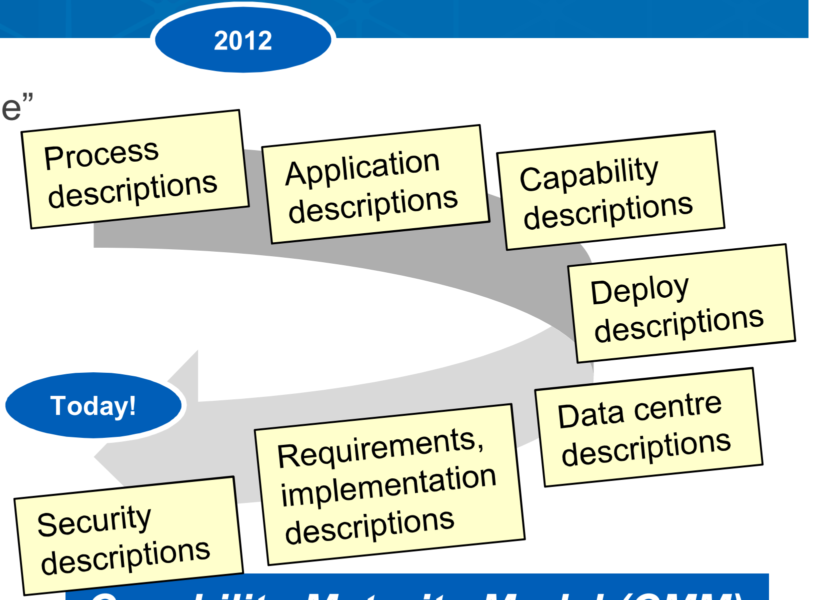
Requirements

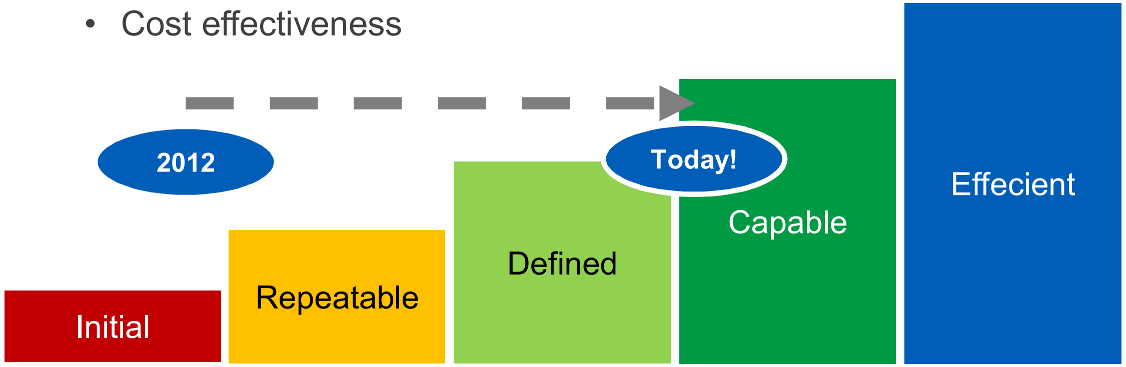
Security

Presentation at-a-glance

Problems – back then… and the maturity aspect

- Requirements analysis “per assignment” and not “as a whole”
  - Go more Agile, sometimes Waterfalling
- Unstructured modelling
  - Napkin models (Servett-modeller)
  - Different formats and distributed storage
  - Difficult to analyze - Focus on solution architecture, not the entire enterprise
- Frameworks like Archimate and TOGAF are too narrow
  - Poor support for traceability to requirements
  - Frameworks don´t collaborate
- Architectural maturity needed in Business, needs coordination
  - Poor tool support for architects
  - Cost effectiveness





Capability Maturity Model (CMM)

Initial - chaotic, ad hoc, individual heroics, the starting point for use of a new or undocumented repeat process.
Repeatable - the process is at least documented sufficiently such that repeating the same steps may be attempted.
Defined - the process is defined/confirmed as a standard business process
Capable - the process is quantitatively managed in accordance with agreed-upon metrics.
Efficient - process management includes deliberate process optimization/improvement.

Wikipedia

Presentation at-a-glance

Enterprise thinking, system thinking

Enterprise architecture

Enterprise thinking ”as a whole”

- Strategic scope
- Architectural capability, skill set, motivation, training
- Standardization (Domain Specific Language)
- Reference libraries (requirements) and architectures (views/diagrams)
- Repository, meta model
- Enterprise transformation plans
- Portfolios for master applications and data
- Policy, principles and quality attributes
- Enterprise activities and analysis
- …

System architecture

System thinking “as boxed”

- Program/project scope
- Project skill, motivation and architectural compliance
- Project reference and requirements libraries
- Project input
- Project architecture, views/diagrams
- Project transformation plan
- Project documentation
- Architecture decisions
- Balancing quality attributes
- Architecture artefact and activities
- Project evaluation, review or analysis
- Technical debt
- …

Meta model

Guiding Diagrams

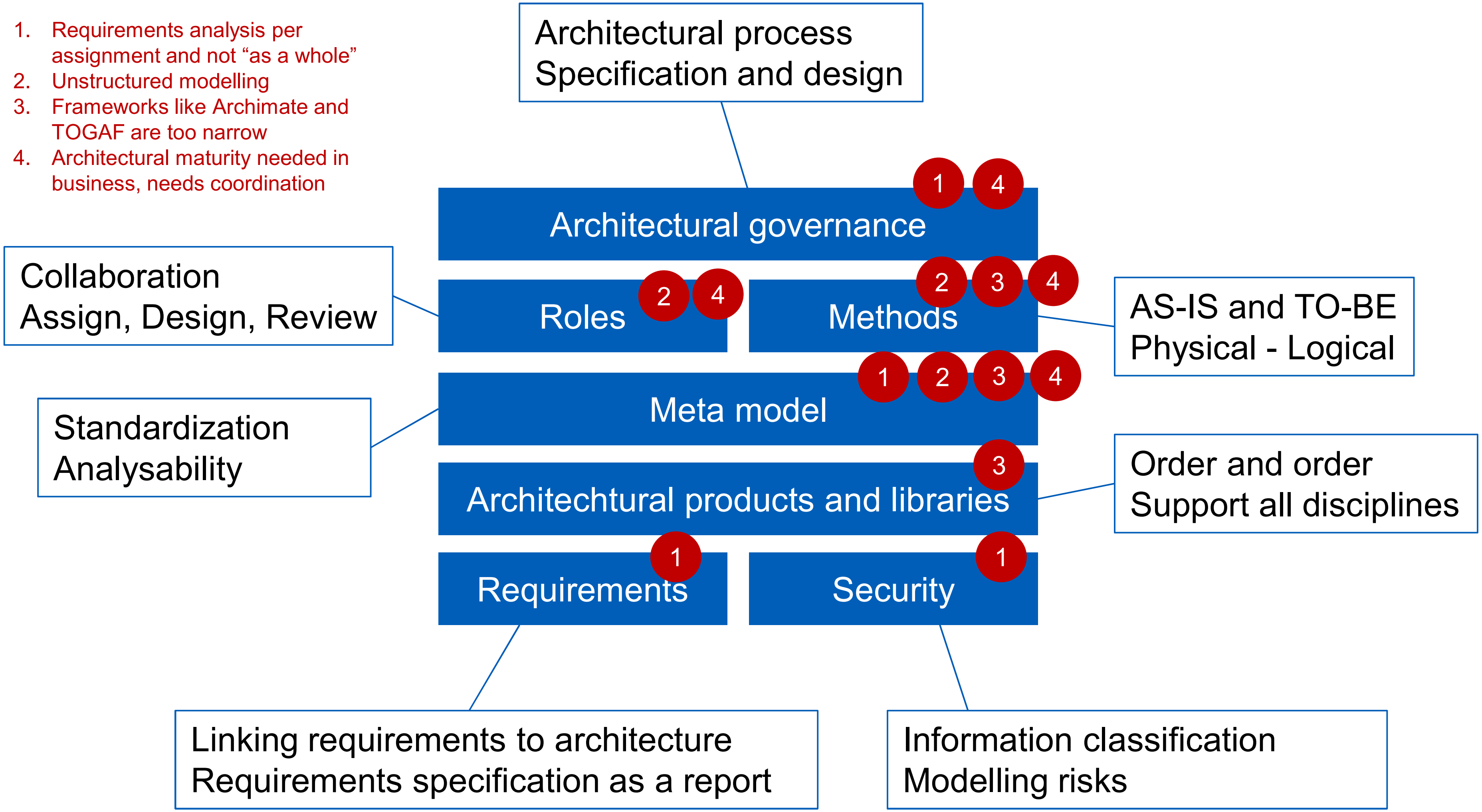
Diagrams

Scetches

Presentation at-a-glance

Focus areas

1. Requirements analysis per assignment and not “as a whole”
2. Unstructured modelling
3. Frameworks like Archimate and TOGAF are too narrow
4. Architectural maturity needed in business, needs coordination



Architectural process
Specification and design

Collaboration
Assign, Design, Review

Standardization
Analysability

AS-IS and TO-BE
Physical - Logical

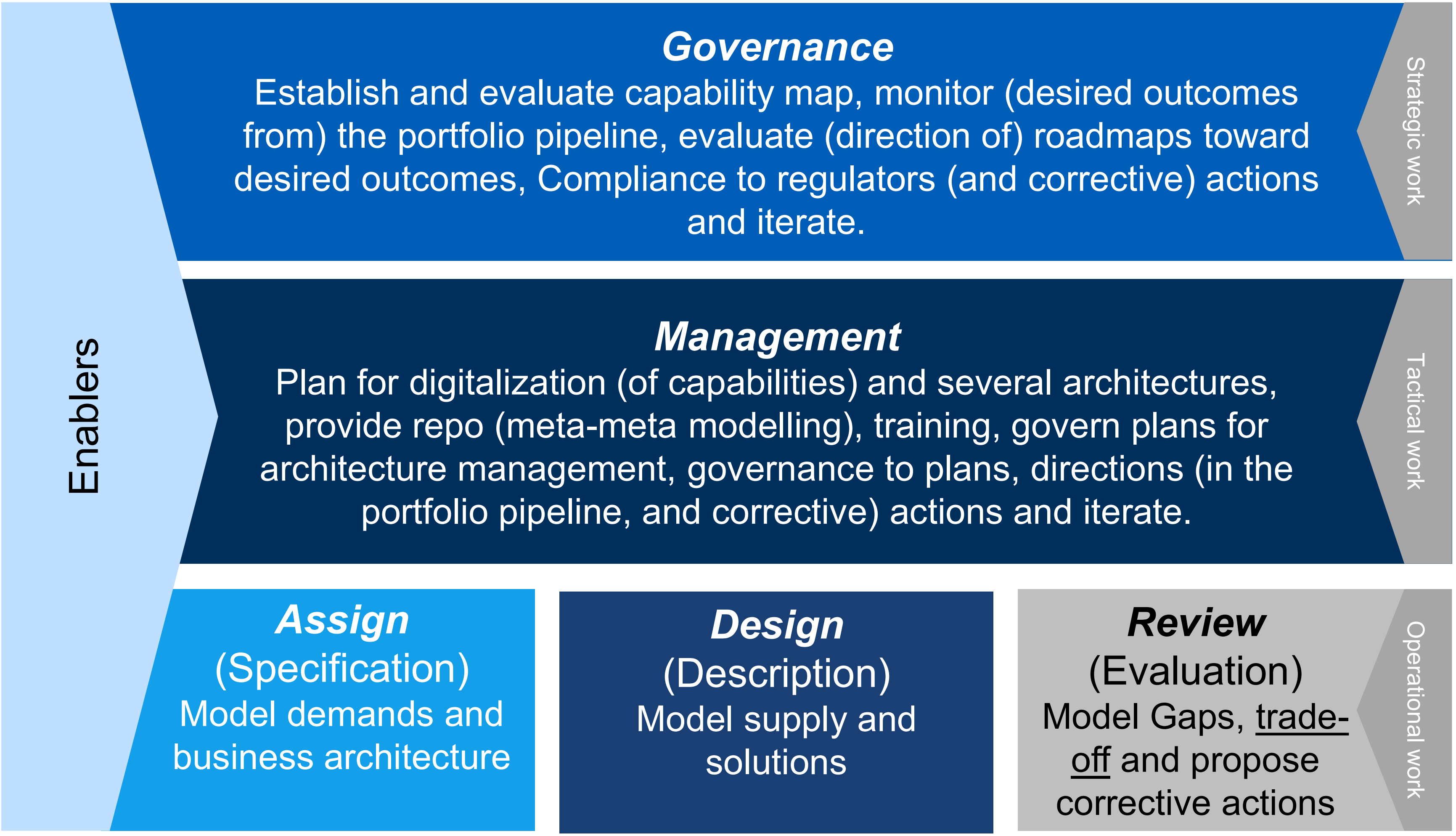
Order and order
Support all disciplines

Linking requirements to architecture
Requirements specification as a report

Information classification
Modelling risks

Architectural governance

Architectural scope strategy, tactical and operation



Enablers

Governance
Establish and evaluate capability map, monitor (desired outcomes from) the portfolio pipeline, evaluate (direction of) roadmaps toward desired outcomes, Compliance to regulators (and corrective) actions and iterate.

Strategic work

Management
Plan for digitalization (of capabilities) and several architectures, provide repo (meta-meta modelling), training, govern plans for architecture management, governance to plans, directions (in the portfolio pipeline, and corrective) actions and iterate.

Tactical work

Assign
(Specification)
Model demands and business architecture

Design
(Description)
Model supply and solutions

Review
(Evaluation)
Model Gaps, trade-off and propose corrective actions

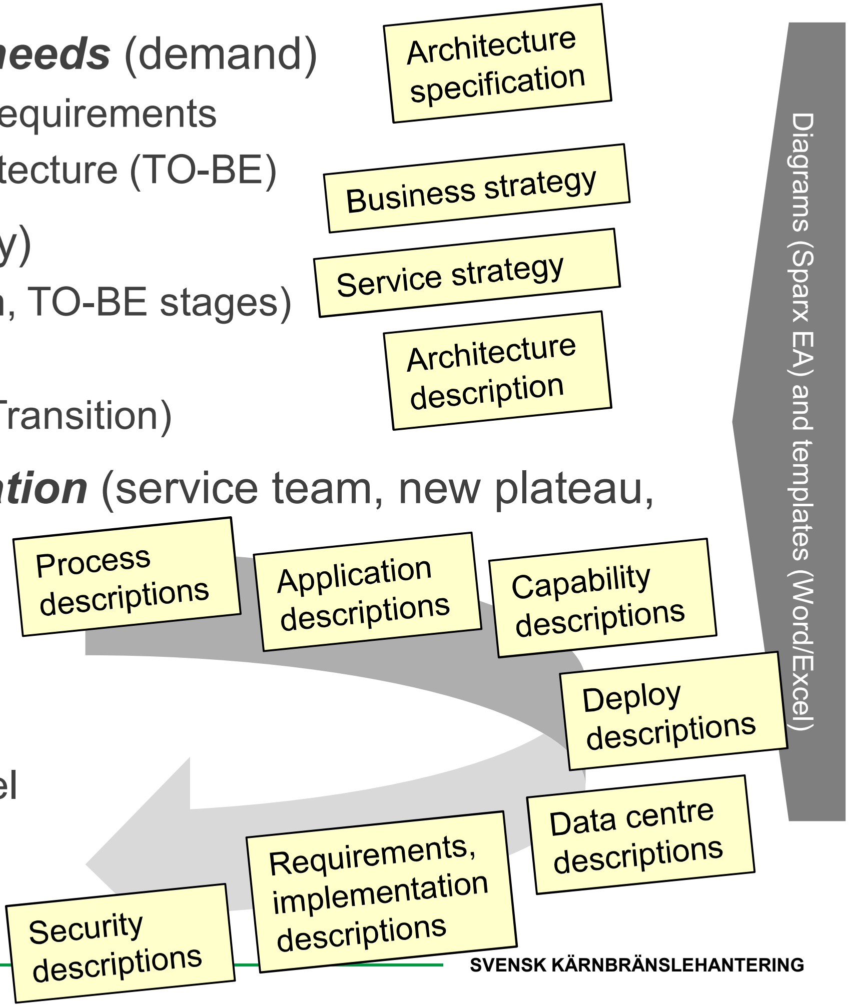
Operational work

Architectural governance

Overview – Core concepts

Standardized artifacts in SOP:s (Standard operational procedures) program/project control and development model

- ***The architecture specification of business needs*** (demand)
  - Achitectural analysis of business architecture and requirements
  - Focus on identifying GAPs, stages and target architecture (TO-BE)
- ***Supplier's description of the solution*** (supply)
  - Solution architecture (Hi-level and Low-level design, TO-BE stages)
  - Focus on RFI and RFQ (Time, Money and Scope)
  - Focus on solution architecture to extinguish GAP (Transition)
- ***Technical documentation of the implementation*** (service team, new plateau, AS-IS)
- ***Toll-gated approach*** with architectural review
  - TG 0, Business case/Investment decision
  - TG 2, Choice of alternatives, Hi-level
  - TG 4, Decision on introduction/realization, Low-level



Architectural governance

Architectural scope and capabilities according to ISO/IEC/IEEE 42020

Conceptualization

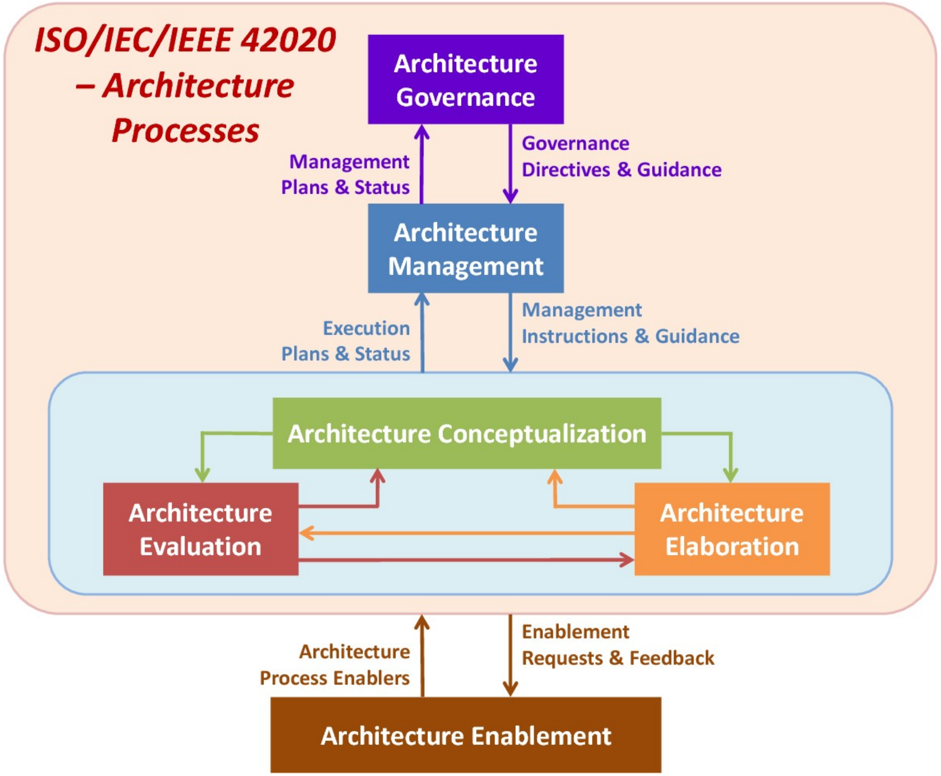
- ***Assign* –** Problem and innovation space definition, architecture objectives, critical success criteria, solution space definition

Elaboration

- ***Design*** – Architecture models and views/diagrams, Demands architecture specifications and Supplys architecture descriptions (Hi-level or Low-level design)

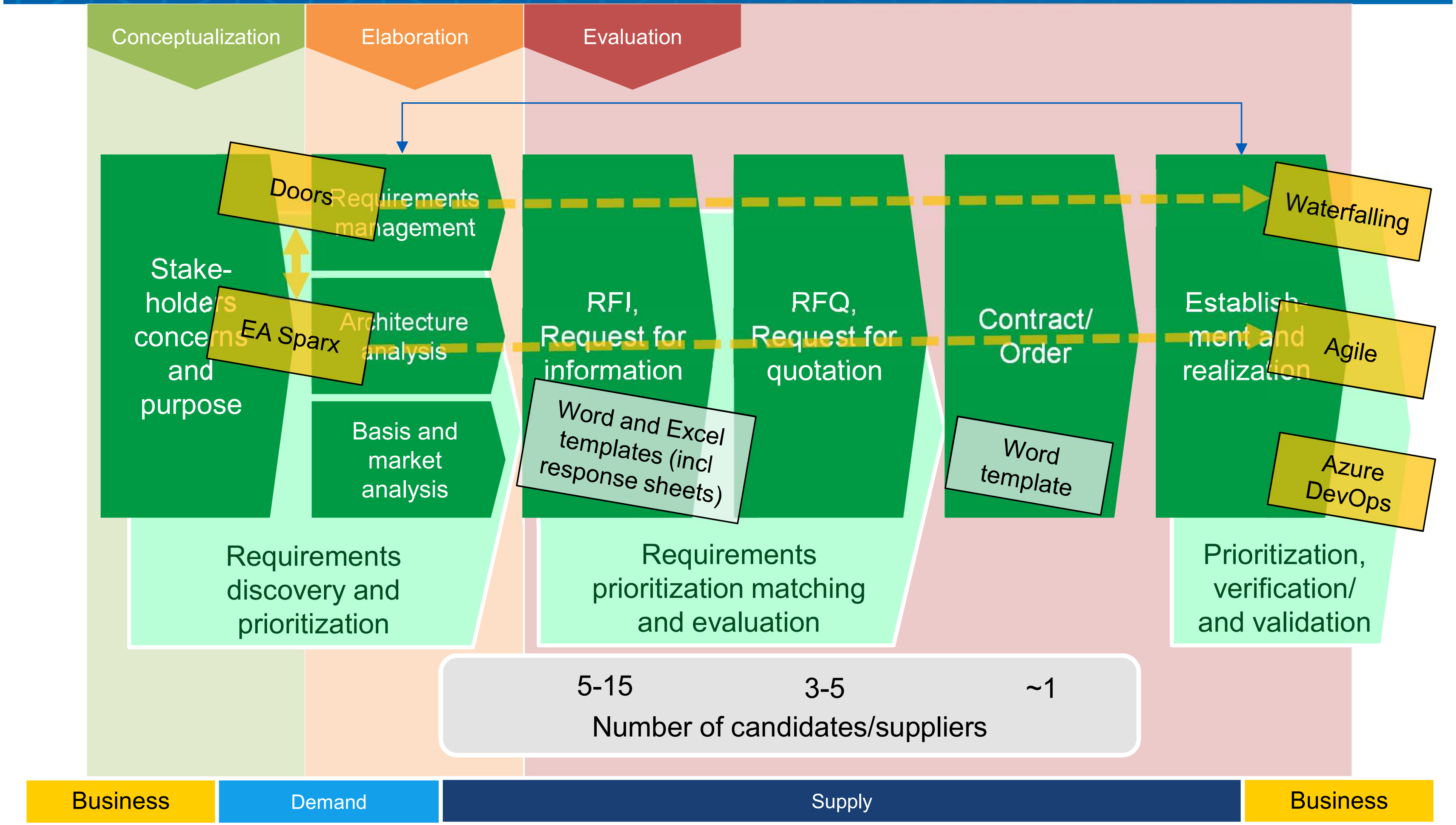
Evaluation

- ***Review* –** Evaluation results, evaluation findings and recommendations


ISO/IEC/IEEE 42020 – Architecture Processes
Architecture Governance
Management Plans & Status
Governance Directives & Guidance
Architecture Management
Execution Plans & Status
Management Instructions & Guidance
Architecture Conceptualization
Architecture Evaluation
Architecture Elaboration
Architecture Process Enablers
Enablement Requests & Feedback
Architecture Enablement

Architectural governance

Delivery tactics – Architectural work, simplified process (part of our PROPS)


Conceptualization
Elaboration
Evaluation
Stake-holders concerns and purpose
Requirements management
Architecture analysis
Basis and market analysis
RFI, Request for information
RFQ, Request for quotation
Contract/ Order
Establish-ment and realization
Doors
EA Sparx
Word and Excel templates (incl response sheets)
Word template
Waterfalling
Agile
Azure DevOps
Requirements discovery and prioritization
Requirements prioritization matching and evaluation
Prioritization, verification/ and validation
5-15
3-5
~1
Number of candidates/suppliers
Business
Demand
Supply
Business

Architectural roles

The architects and enterprise maturity (significance, skill set, IASA 5 pillars)

SKB

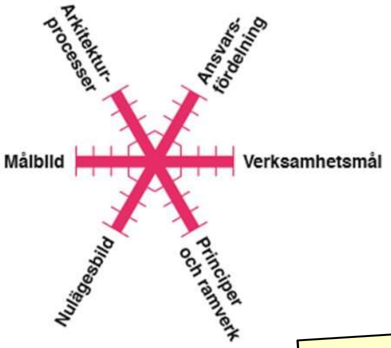
Enterprise premises

- Distribution of responsibilities
- Business goals
- Principles, frameworks and methods
- Current image
- Target image
- Architectural processes

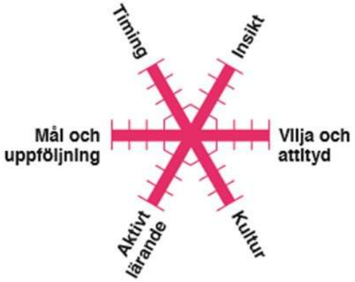
Architecture excellence

- Insight
- "Will" and attitude
- Culture and knowledge
- Active learning
- Evaluation and quality assurance
- Timing

Förutsättningar



Skickligheter



Know your CMM level 1-5

Business Technology Strategy

- Business Fundamentals
- Strategy Rationalization and Development
- Industry Analysis
- Business Valuation
- Investment Prioritization and Planning
- Requirements Discovery and Constraints Analysis
- Compliance
- Business Architecture Methods & Tools
- Decision Support
- Knowledge Management

IT Environment

- Technical Project Management
- Asset Management
- Change Management
- Infrastructure
- Application Development
- Governance
- Testing Methods, Tools, and Techniques
- Platforms and Frameworks

Design Skills

- Requirements Modeling
- Architecture Description
- Decomposition and Reuse
- Design Methodologies and Processes
- Design Patterns and Styles
- Design Analysis and Testing
- Traceability Throughout the Lifecycle
- Views & Viewpoints
- The Whole Systems Design

Human Dynamics

- Managing the Culture
- Customer Relations
- Leadership and Management
- Peer Interaction
- Collaboration and Negotiation
- Presentation Skills
- Writing Skills

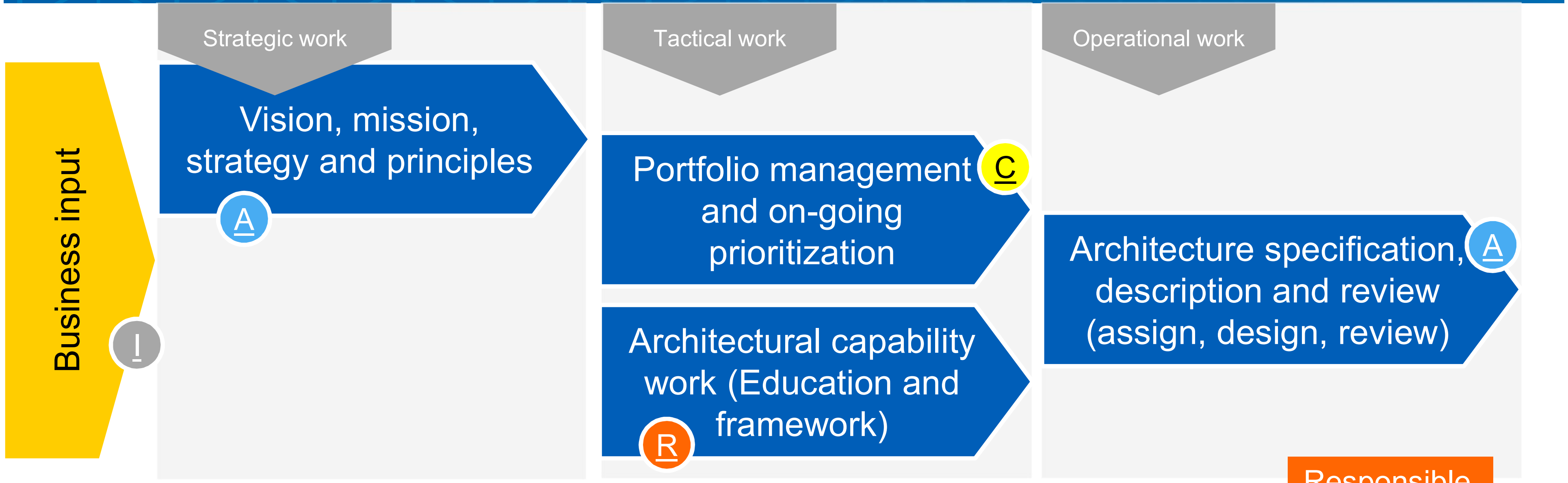
Quality Attributes

- Balancing and Optimizing Quality Attributes
- Manageability, Maintainability, Supportability, Extensibility, and Flexibility
- Monitoring and Management
- Performance, Reliability, Availability, Scalability
- Security
- Usability, Localization, Accessibility, Personalization/Customizability
- Packaging, Delivery, Post Deployment

IASA Self assessment

Architectural roles

Core activities and responsibility assignment



Business input I

Strategic work

Vision, mission, strategy and principles A

Tactical work

Portfolio management and on-going prioritization C

Architectural capability work (Education and framework) R

Operational work

Architecture specification, description and review (assign, design, review) A

Responsible

Accountable

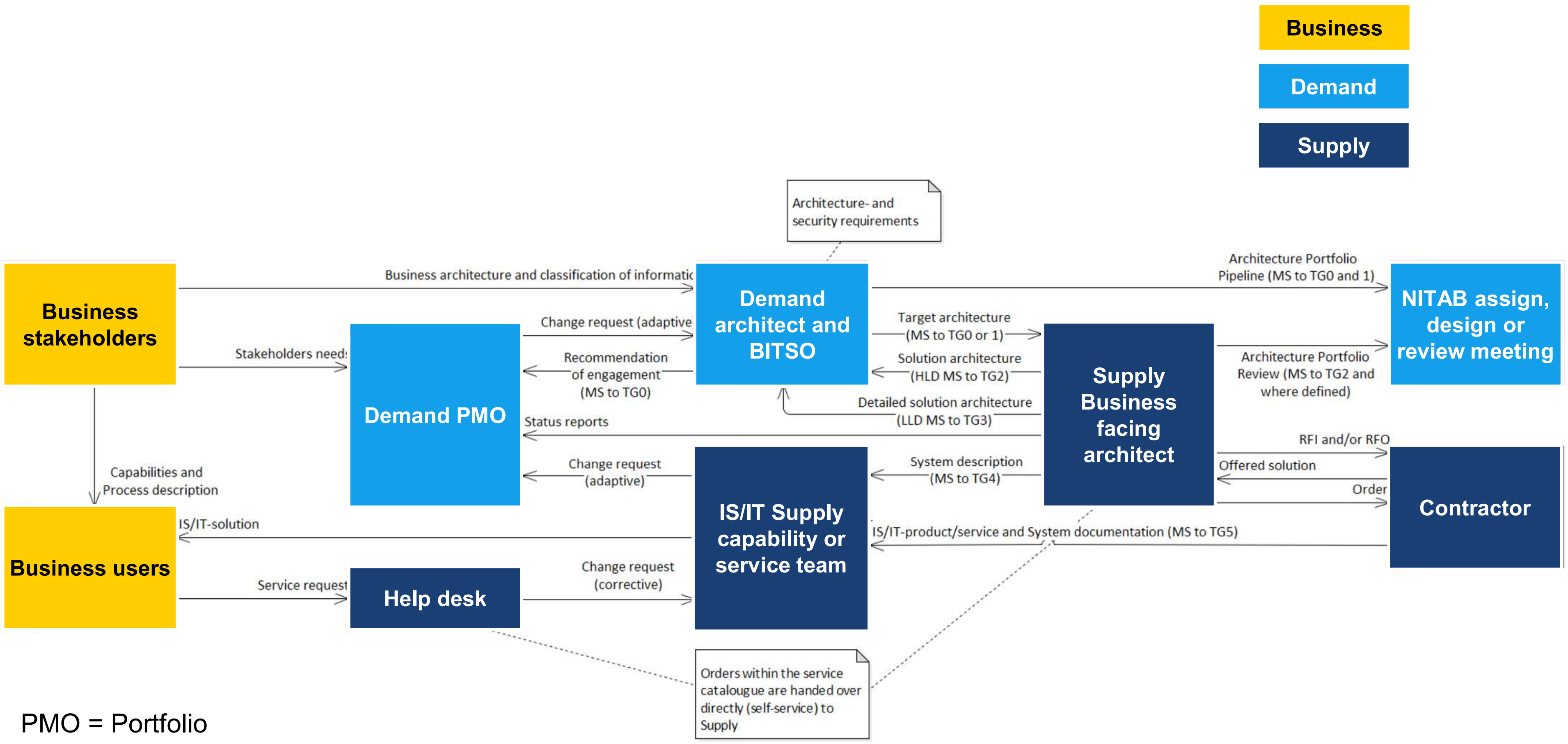
Consulted

Informed

- IT support business input with a stakeholders view – Example; Line of business, process owners, regulatory interpretation managers, information security managers and BI (Business intelligence).
- Various architect roles interacts (next slide) in all activities. Applies to both demand and supply.
- Review and feedback is important for the continual improvement of both activities and roles

Architectural roles

Collaboration sequence diagram and artefacts

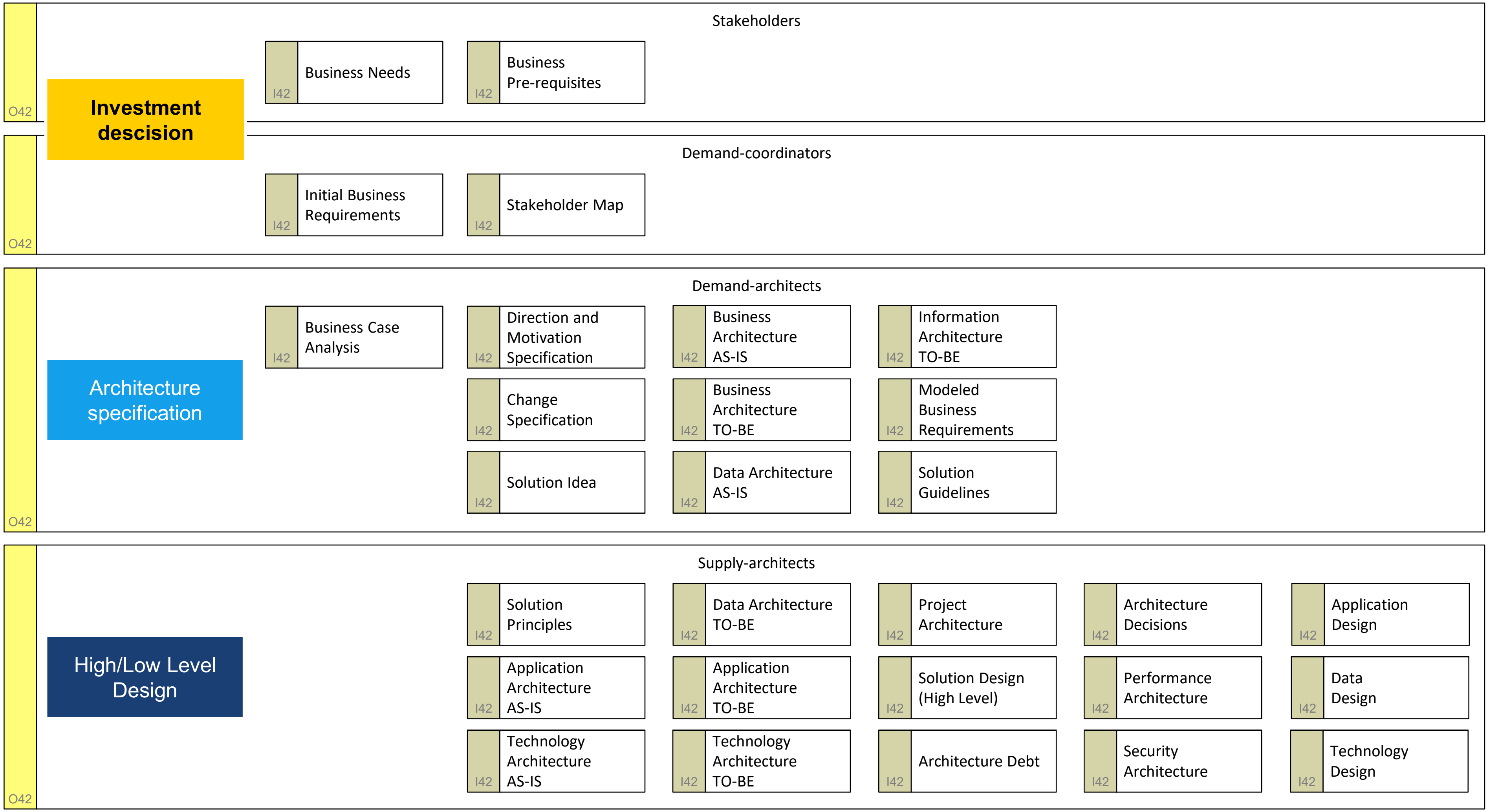


Business
Demand
Supply
Architecture- and security requirements
Business stakeholders
Business architecture and classification of informatic
Demand architect and BITSO
Architecture Portfolio Pipeline (MS to TG0 and 1)
NITAB assign, design or review meeting
Stakeholders needs
Demand PMO
Change request (adaptive
Recommendation of engagement (MS to TG0)
Target architecture (MS to TG0 or 1)
Solution architecture (HLD MS to TG2)
Detailed solution architecture (LLD MS to TG3)
Supply Business facing architect
Architecture Portfolio Review (MS to TG2 and where defined)
Status reports
RFI and/or RFO
Offered solution
Order
Contractor
Capabilities and Process description
Change request (adaptive)
System description (MS to TG4)
IS/IT Supply capability or service team
Business users
IS/IT-solution
IS/IT-product/service and System documentation (MS to TG5)
Service request
Help desk
Change request (corrective)
Orders within the service catalougue are handed over directly (self-service) to Supply

PMO = Portfolio
NITAB = Nuclear IT Architect Board
BITSO = Business Information Technology Security Officer
Capability teams runs PM3
Service teams runs ITIL

Architectural roles

Division of responsibilities to architectural core artifacts



Stakeholders

O42

Investment descision

I42 Business Needs

I42 Business Pre-requisites

Demand-coordinators

O42

I42 Initial Business Requirements

I42 Stakeholder Map

Demand-architects

O42

Architecture specification

I42 Business Case Analysis

I42 Direction and Motivation Specification

I42 Business Architecture AS-IS

I42 Information Architecture TO-BE

I42 Change Specification

I42 Business Architecture TO-BE

I42 Modeled Business Requirements

I42 Solution Idea

I42 Data Architecture AS-IS

I42 Solution Guidelines

Supply-architects

O42

High/Low Level Design

I42 Solution Principles

I42 Data Architecture TO-BE

I42 Project Architecture

I42 Architecture Decisions

I42 Application Design

I42 Application Architecture AS-IS

I42 Application Architecture TO-BE

I42 Solution Design (High Level)

I42 Performance Architecture

I42 Data Design

I42 Technology Architecture AS-IS

I42 Technology Architecture TO-BE

I42 Architecture Debt

I42 Security Architecture

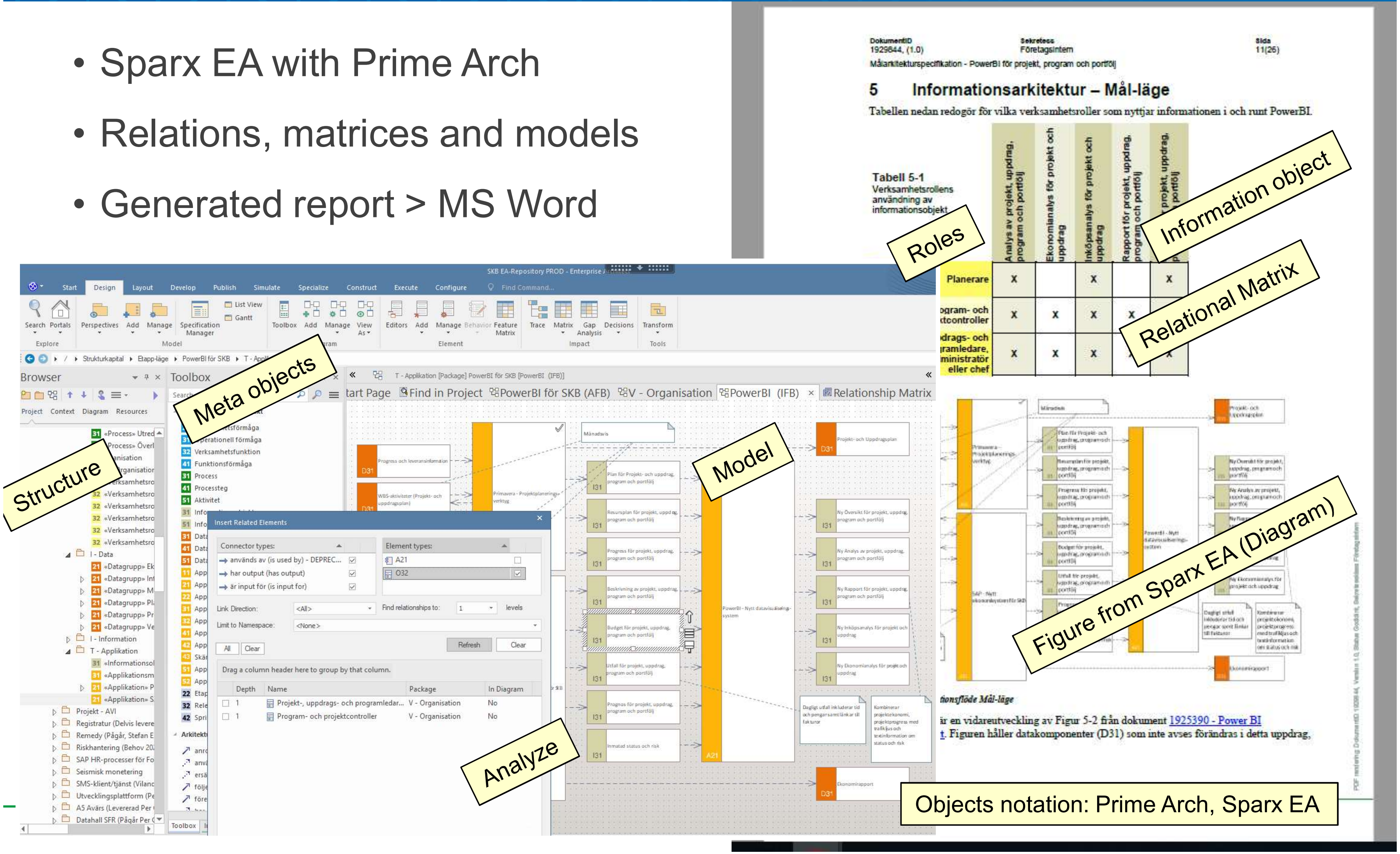
I42 Technology Design

Objects notation: Prime Arch, Sparx EA

Architectural role

Modelling model in Sparx EA

- Sparx EA with Prime Arch
- Relations, matrices and models
- Generated report > MS Word



Depth	Name	Package	In Diagram
1	Projekt-, uppdrags- och programledar...	V - Organisation	No
1	Program- och projektcontroller	V - Organisation	No

DokumentID 1929844, (1.0) Sekretess Företagsintern Sida 11(26)

Målarkitekturspecifikation - PowerBI för projekt, program och portfölj

5 Informationsarkitektur – Mål-läge

Tabellen nedan redogör för vilka verksamhetsroller som nyttjar informationen i och runt PowerBI.

Tabell 5-1 Verksamhetsrollens användning av informationsobjekt

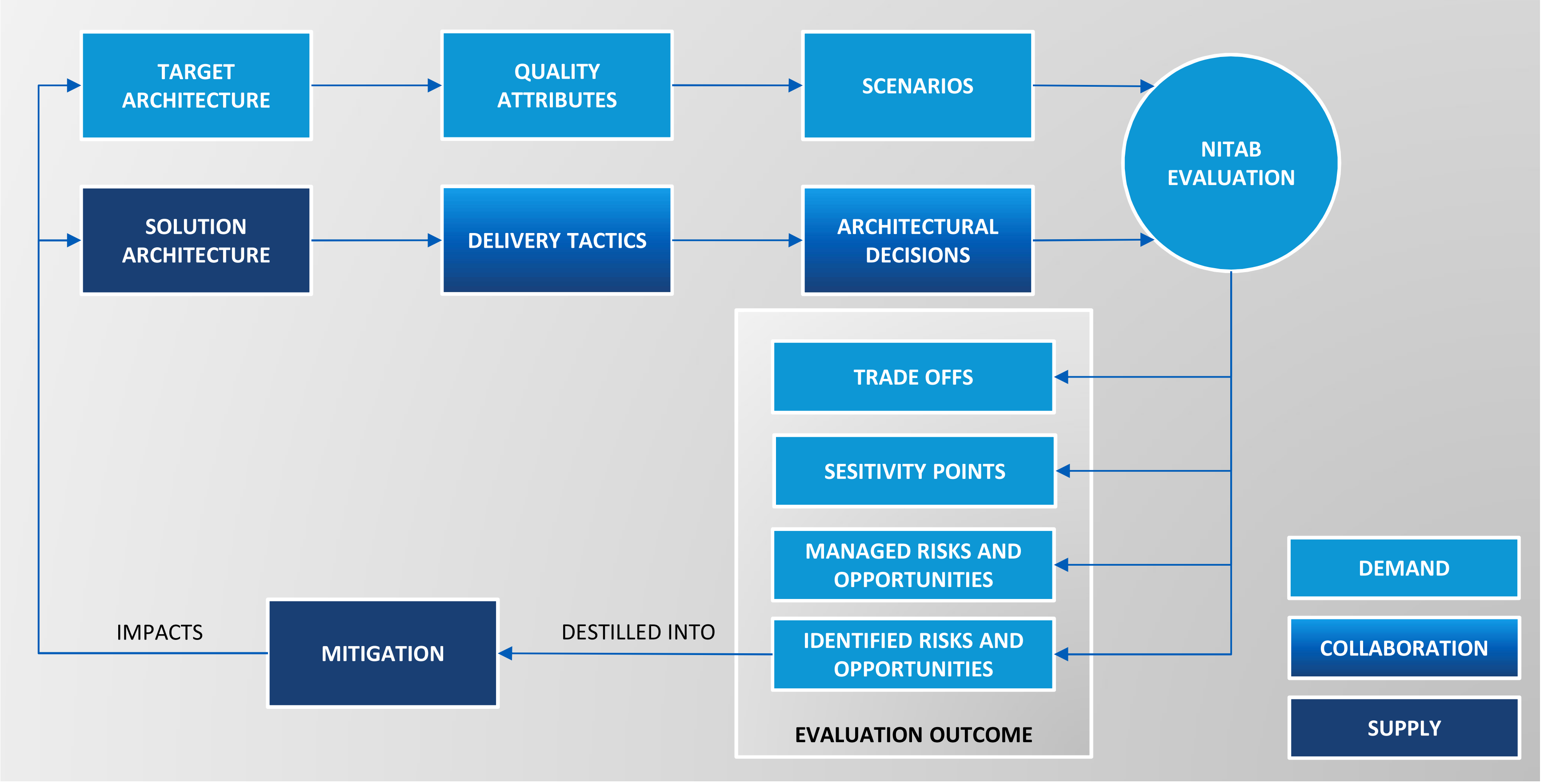
	Analys av projekt, uppdrag, program och portfölj	Ekonomianalys för projekt och uppdrag	Inköpsanalys för projekt och uppdrag	Rapport för projekt, uppdrag, program och portfölj	projekt, uppdrag, portfölj
Planerare	X		X		X
ogram- och ktcontroller	X	X	X	X	
drags- och ramledare, ministratör eller chef	X	X	X		X

tionsflöde Mål-läge

r en vidareutveckling av Figur 5-2 från dokument 1925390 - Power BI t. Figuren håller datakomponenter (D31) som inte avses förändras i detta uppdrag.

Architectural methods

ATAM – Architectural Trade-off analysis method

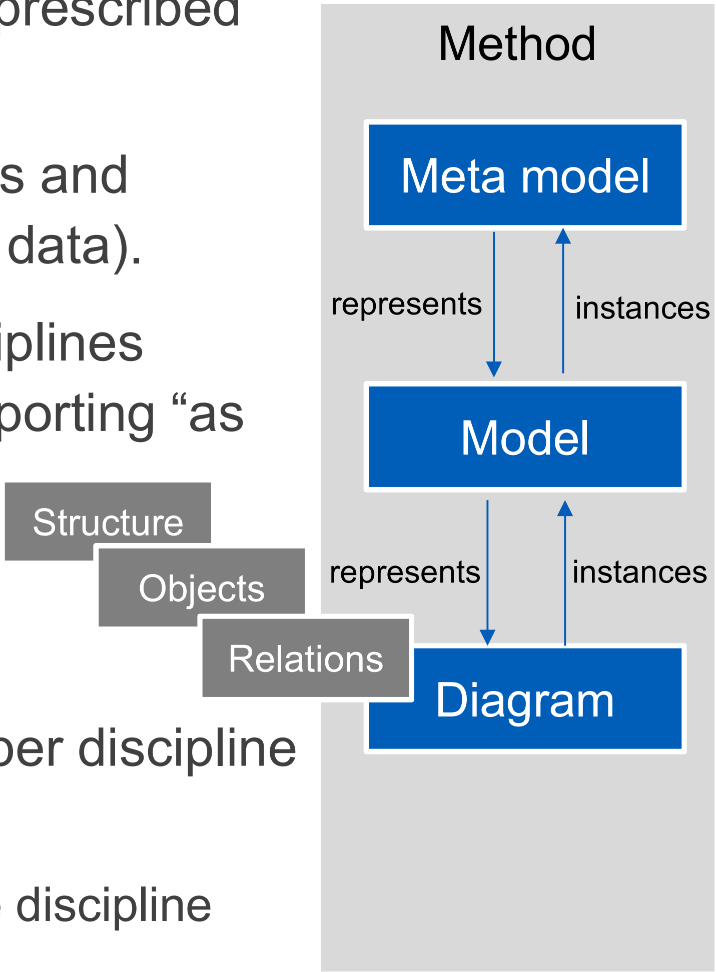


Architectural methods

Model driven engineering

A method is the pursuit of knowledge from a prescribed process when completing a task.

- ***Sparx EA*** – A method refers to specific tools and procedures (you use to collect and analyze data).
- ***Define meta model*** – Prime Arch with disciplines (domains) and levels of detailing (LoD) supporting "as is" and "to be" (CAB engagement)
- ***Host the repo (model)*** – in Sparx EA (capability service ADI)
- ***Model views (diagram)*** – standardization per discipline
  - ***Composition*** – that reveals detailing
  - ***Collaboration*** – interacting as flows within the discipline
  - ***Contextual*** – interacting with other disciplines

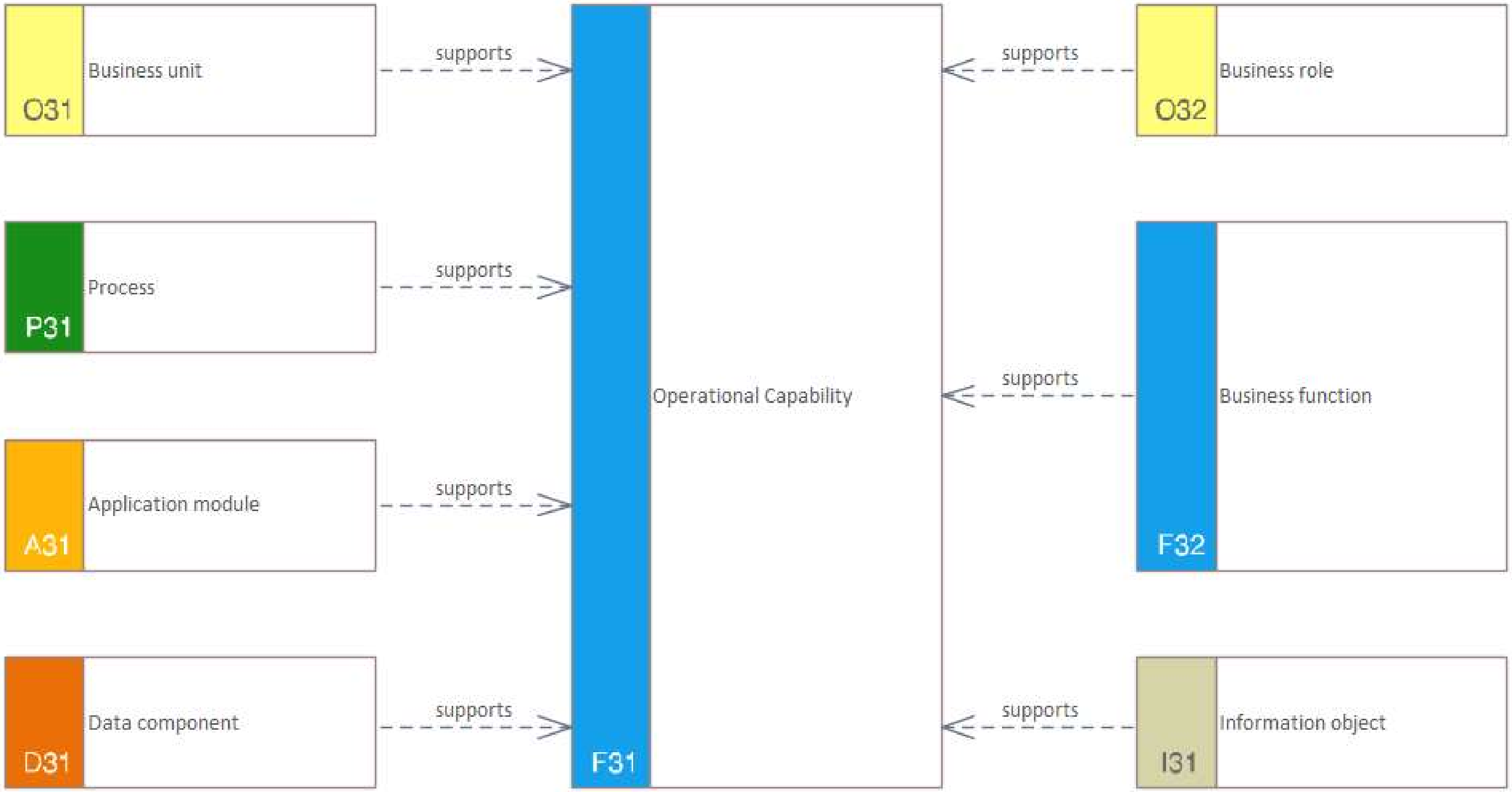


Architectural methods

The operational capability "short cut" to functional requirements

As is
As implemented

To be
As required



Contextual view on an Operational capability. Objects notation: Prime Arch, Sparx EA

Architectural methods

Information och Data – “To be”, “As is” and “Transition”

INFORMATION

- What information **should** we have?
- What information **should** be transferred between our applications?
- What information **should** an "information package" (which is shared between different Business functions) consist of?
- How **should** business's important concepts be defined?
- How **should** the information be structured?

TO BE

DATA

- What data do we have?
- What data is transferred between our applications?
- What does a "data packet" (which is transferred between applications) consist of?
- How is all data defined in our applications?
- How is all the data structured?

AS IS

THE GAP BETWEEN INFORMATION AND DATA

- How well does the data landscape **match** the company's information needs?
- What can be **reused** for the target architecture?

TRANSITION

Architectual methods

Logical and physical application – “To be”, “As is” and “Transition”

LOGICAL APPLICATION

- What applications **should** we have?
- What functions **should** be supported by the specific application?
- What applications **should** collaborate?
- What data **should** be defined in our applications?
- How **should** the specific application relate to other parts of the enterprise?

TO BE

PHYSICAL APPLICATION

- What applications do we have?
- What functions are supported by the specific application?
- What applications collaborate?
- What data defined in our applications?
- How does the specific application relate to other parts of the enterprise?

AS IS

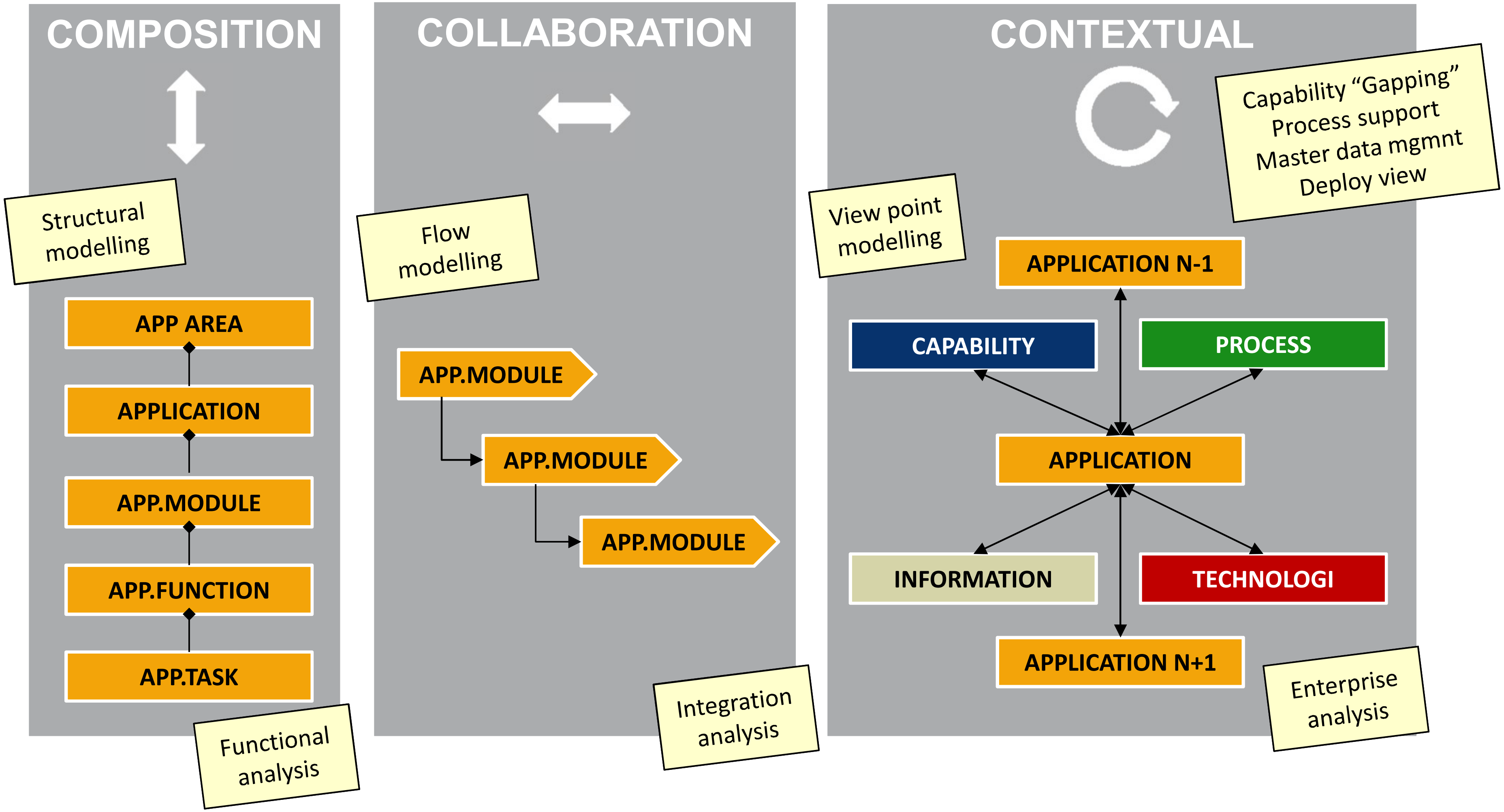
THE GAP BETWEEN LOGICAL AND PHYSICAL APPLICATION

- How well does the application **match** the target architecture, digitalizing capabilities?
- What can be **reused** for the target architecture?

TRANSITION

Architectural methods

Diagram perspectives – Example from the application dimension



Architectural methods

Diagram perspectives – Meta examples

PHYSICAL APPLICATION

COMPOSITION

Is part of/consists of, input/output

What applications do we have?

What functions are supported by the specific application?

COLLABORATION

Follows/Precedes, Sends/Receives, Performs

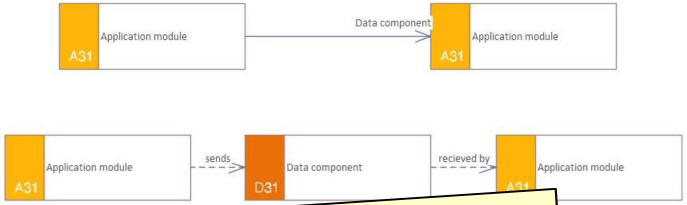
What applications collaborate?

What data defined in our applications?

CONTEXTUAL

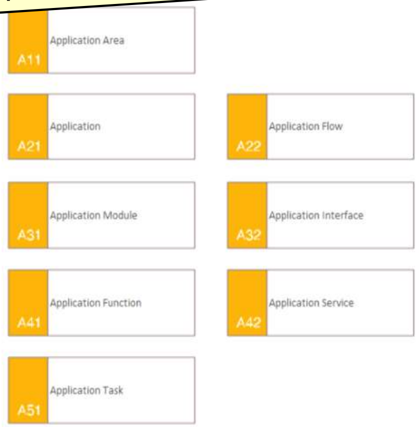
Is part of/consists of, input/output

How does the specific application relate to other parts of the enterprise?

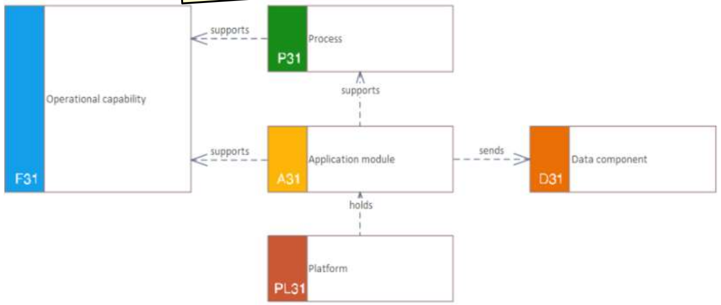


Collaborations view on applications

Composition view on applications



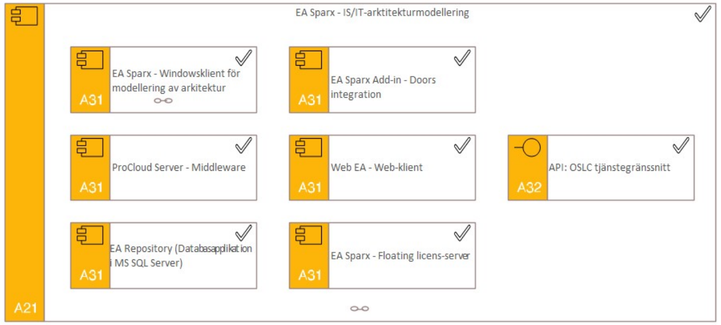
Contextual view on an application



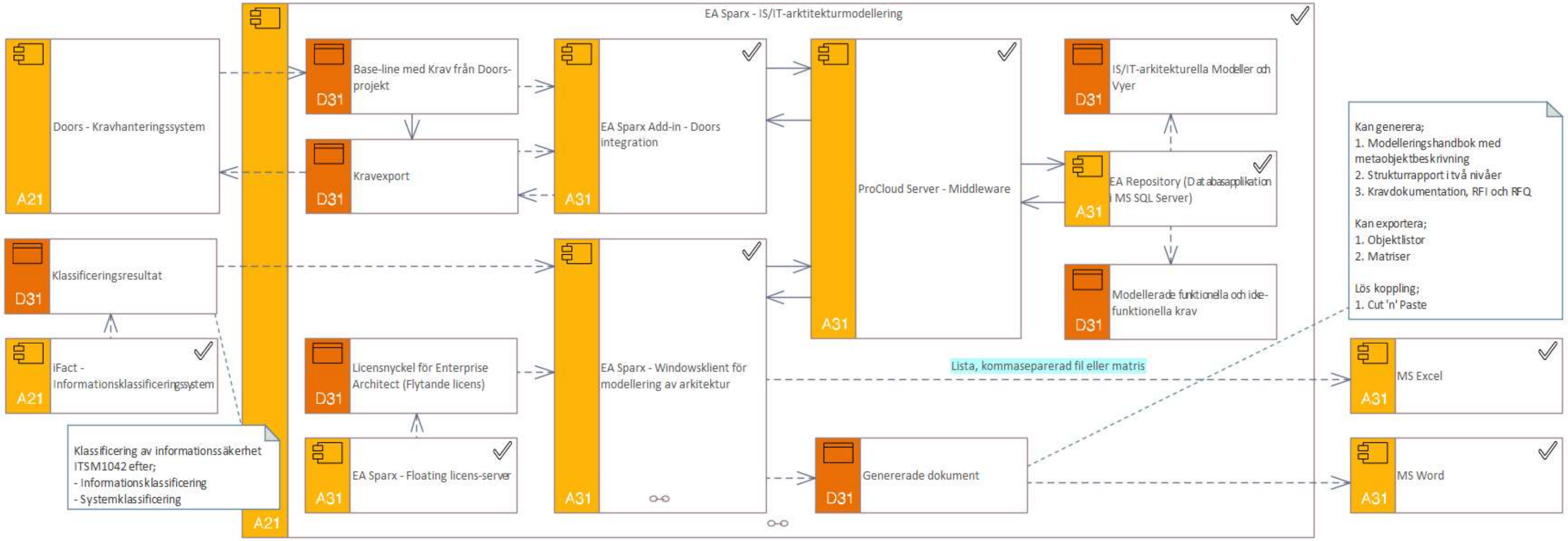
Architectural methods

Sparx EA, Examples Application description and Application flow description

COMPOSITION


EA Sparx - IS/IT-arktitekturmodellering
A21
EA Sparx - Windowsklient för modellering av arkitektur
A31
EA Sparx Add-in - Doors integration
A31
ProCloud Server - Middleware
A31
Web EA - Web-klient
A31
API: OSLC tjänstegränssnitt
A32
EA Repository (Databasapplikation i MS SQL Server)
A31
EA Sparx - Floating licens-server
A31

COLLABORATION


EA Sparx - IS/IT-arktitekturmodellering
A21
Doors - Kravhanteringssystem
A21
Base-line med Krav från Doors-projekt
D31
Kravexport
D31
EA Sparx Add-in - Doors integration
A31
ProCloud Server - Middleware
A31
IS/IT-arkitekturella Modeller och Vyer
D31
EA Repository (Databasapplikation i MS SQL Server)
A31
Modellerade funktionella och icke-funktionella krav
D31
Klassificeringsresultat
D31
iFact - Informationsklassificeringssystem
A21
Licensnyckel för Enterprise Architect (Flytande licens)
D31
EA Sparx - Floating licens-server
A31
EA Sparx - Windowsklient för modellering av arkitektur
A31
Lista, kommaseparerad fil eller matris
Genererade dokument
D31
MS Excel
A31
MS Word
A31

Klassificering av informationssäkerhet ITSM1042 efter;
- Informationsklassificering
- Systemklassificering

Kan generera;
1. Modelleringshandbok med metaobjektbeskrivning
2. Strukturrapport i två nivåer
3. Kravdokumentation, RFI och RFQ

Kan exportera;
1. Objektlistor
2. Matriser

Lös koppling;
1. Cut 'n' Paste

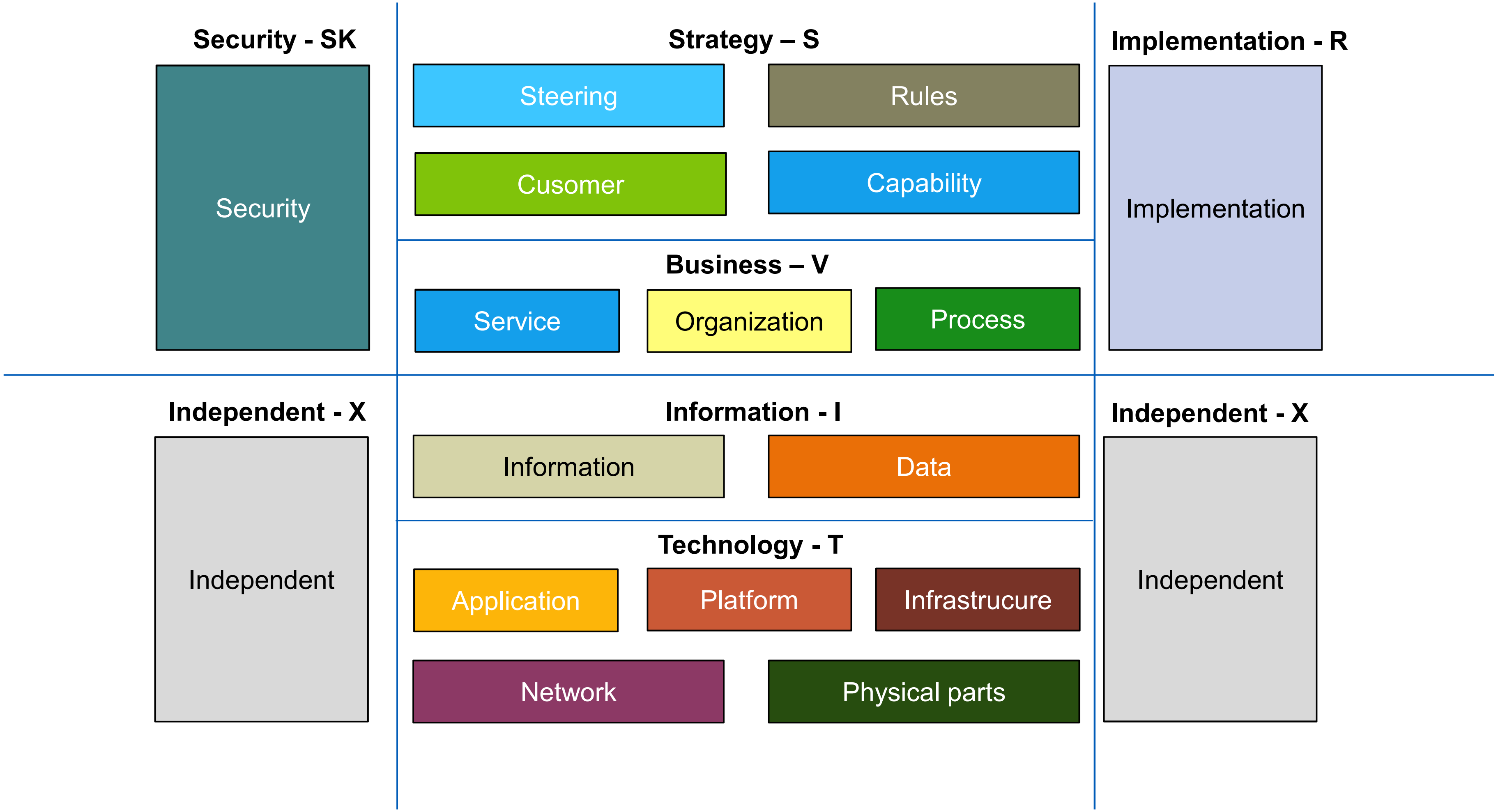
CONTEXTUAL

Contextual view is not included in this presentation

Objects notation: Prime Arch, Sparx EA

Meta model

Prime Arch as an EA-framework – and it´s dimensions



Security - SK

Security

Strategy – S

Steering

Rules

Cusomer

Capability

Business – V

Service

Organization

Process

Implementation - R

Implementation

Independent - X

Independent

Information - I

Information

Data

Technology - T

Application

Platform

Infrastrucure

Network

Physical parts

Independent - X

Independent

Meta model

Prime Arch as an EA-framework

Model-driven software development gives:

- ***A "common" language***, and (of course) the structure and methods through the DSL, domain-specific modeling languages

  Better discussions with stakeholders using disciplines and level of detailing (LoD)
  Structured growth with Simple and relevant diagrams

- ***Validation of modelling*** against the constraints in the metamodel

  Easy to do right with quick link
  Learn while modelling
  "System thinking" and "enterprise thinking"

- ***Diagram-to-diagram transformations*** in the repo or between different repos

  Re-use, manage models across the lifecycle,
  Support other notations like Archimate, BPMN, UML eg

- ***Tool integration*** based on the metamodel and adapted to the respective discipline

  Export data, classes and diagram to other tools like
  DOORS, Word, Azure DevOps eg

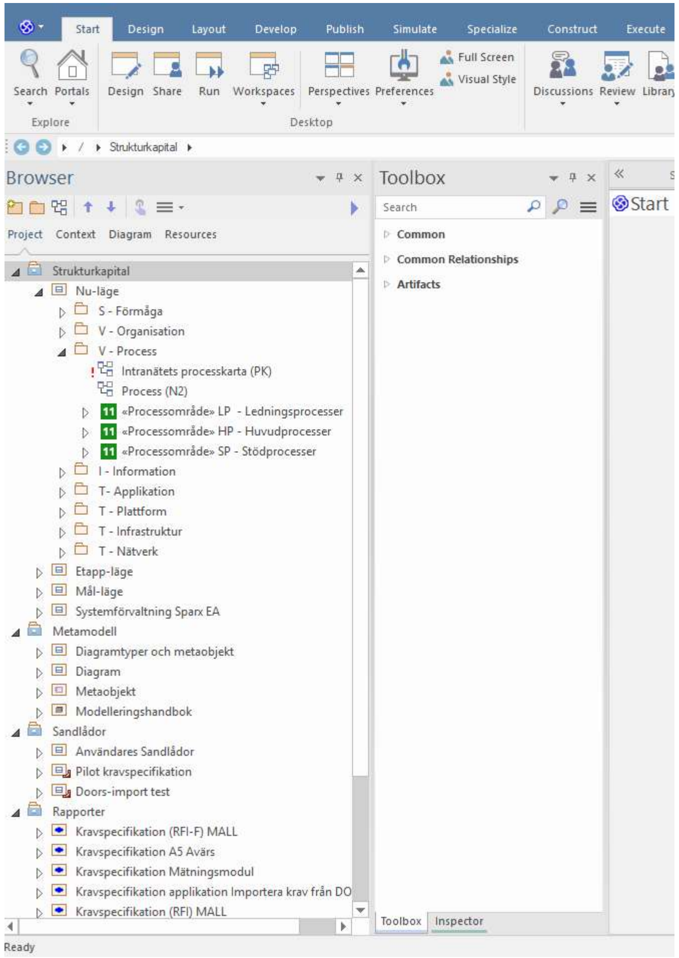
- ***Code generation*** reapetable that refer to the DSL

  Kick-start development

Architecture products and libraries

Whats in it?

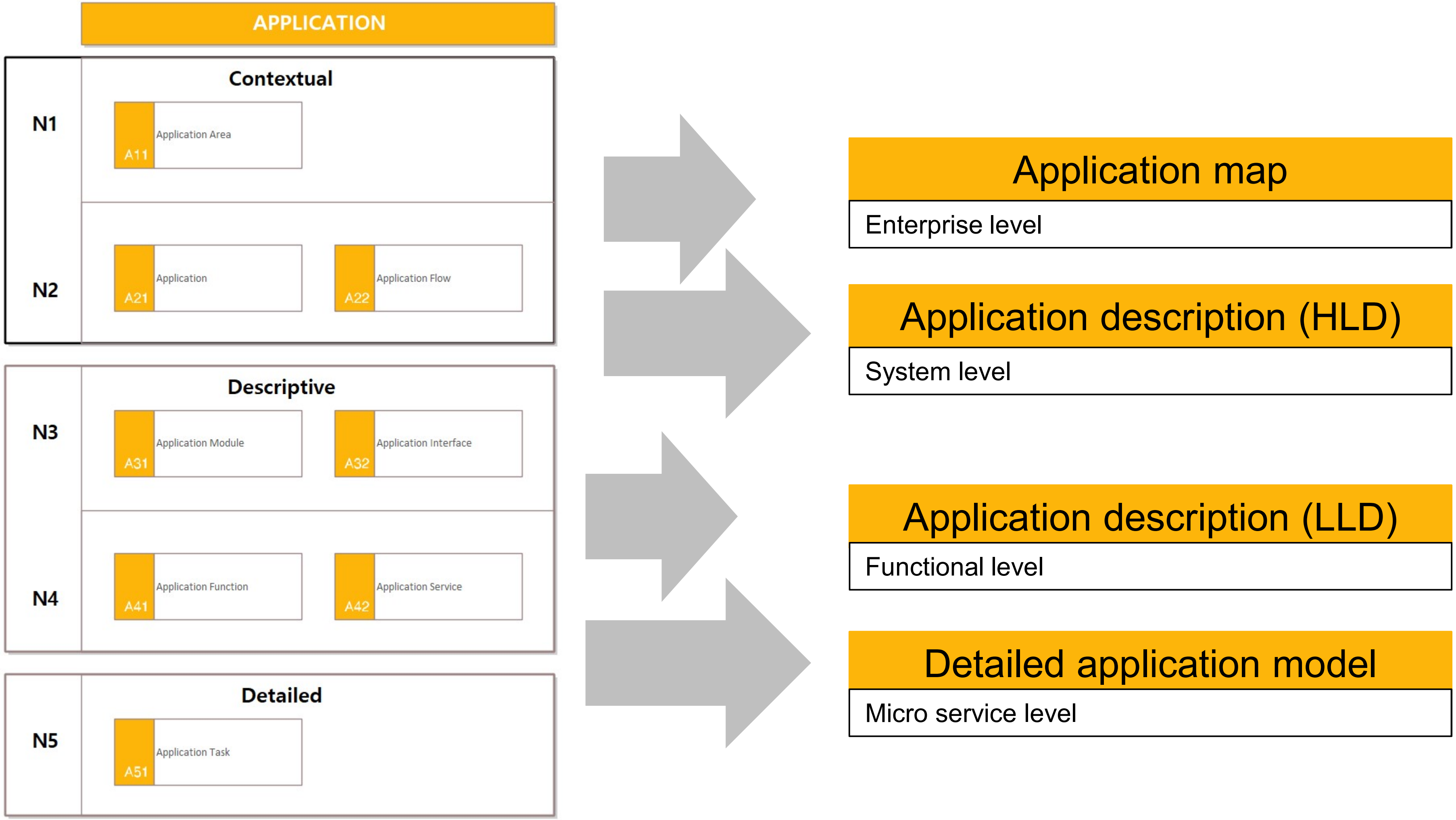
- Package structure
  - As is – Transitional – To Be
  - Object collections with level of detailing
  - Sand boxes (play grounds)
- Meta model
- Object descriptions (including relations)
- Diagrams (ViewTypes)
- Templates (simple and advanced) for generating documentation- or publishing architecture
- Web based training packages (demand and supply)



- Strukturkapital
  - Nu-läge
    - S - Förmåga
    - V - Organisation
    - V - Process
      - Intranätets processkarta (PK)
      - Process (N2)
      - «Processområde» LP - Ledningsprocesser
      - «Processområde» HP - Huvudprocesser
      - «Processområde» SP - Stödprocesser
    - I - Information
    - T- Applikation
    - T - Plattform
    - T - Infrastruktur
    - T - Nätverk
  - Etapp-läge
  - Mål-läge
  - Systemförvaltning Sparx EA
- Metamodell
  - Diagramtyper och metaobjekt
  - Diagram
  - Metaobjekt
  - Modelleringshandbok
- Sandlådor
  - Användares Sandlådor
  - Pilot kravspecifikation
  - Doors-import test
- Rapporter
  - Kravspecifikation (RFI-F) MALL
  - Kravspecifikation A5 Avärs
  - Kravspecifikation Mätningsmodul
  - Kravspecifikation applikation Importera krav från DO
  - Kravspecifikation (RFI) MALL

Architecture products and libraries

Discipline detailed with LoD to support different diagrams (ViewTypes)



Application map

Enterprise level

Application description (HLD)

System level

Application description (LLD)

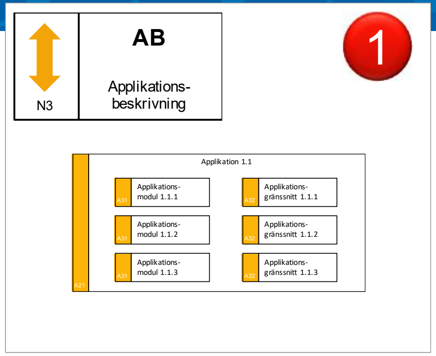
Functional level

Detailed application model

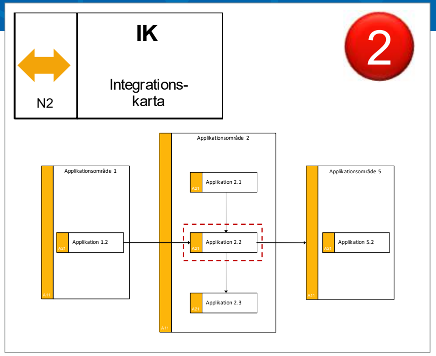
Micro service level

Architecture products and libraries

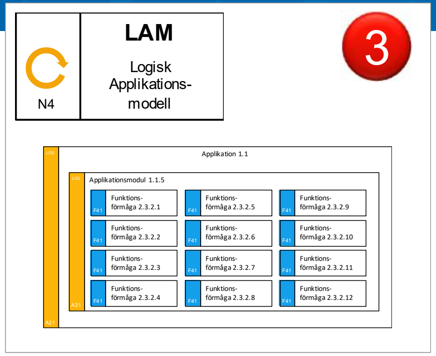
Diagram/ViewType standardization, example from application RFI or RFQ



Description of the application in which the targeted application module is included

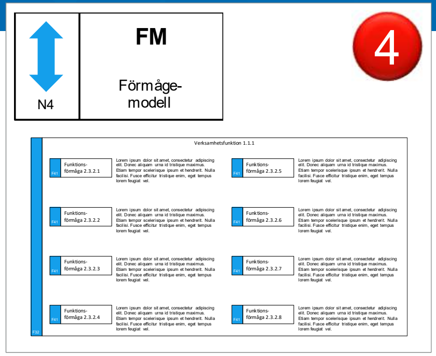


Description of surrounding applications (integrations)

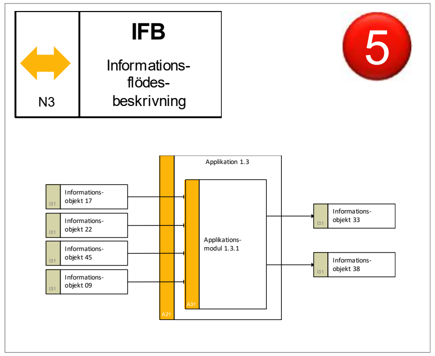


LAM
N4
Logisk Applikations-modell
3
Applikation 1.1
Applikationsmodul 1.1.5
Funktions-förmåga 2.3.2.1
Funktions-förmåga 2.3.2.2
Funktions-förmåga 2.3.2.3
Funktions-förmåga 2.3.2.4
Funktions-förmåga 2.3.2.5
Funktions-förmåga 2.3.2.6
Funktions-förmåga 2.3.2.7
Funktions-förmåga 2.3.2.8
Funktions-förmåga 2.3.2.9
Funktions-förmåga 2.3.2.10
Funktions-förmåga 2.3.2.11
Funktions-förmåga 2.3.2.12

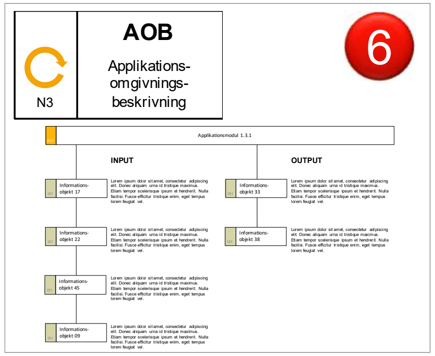
Capabilities (map functional requirements) in the application module



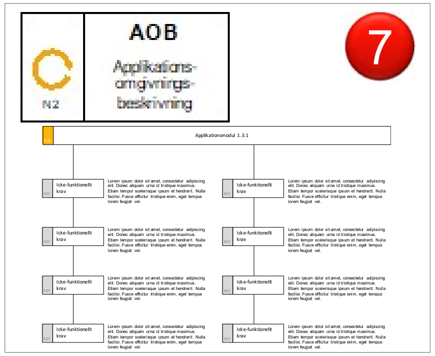
Description of capabilities



Requirements on information (input/output)



Description of information object



The most important non-functional requirements

8

Optional maps, descriptions or models (even textual)

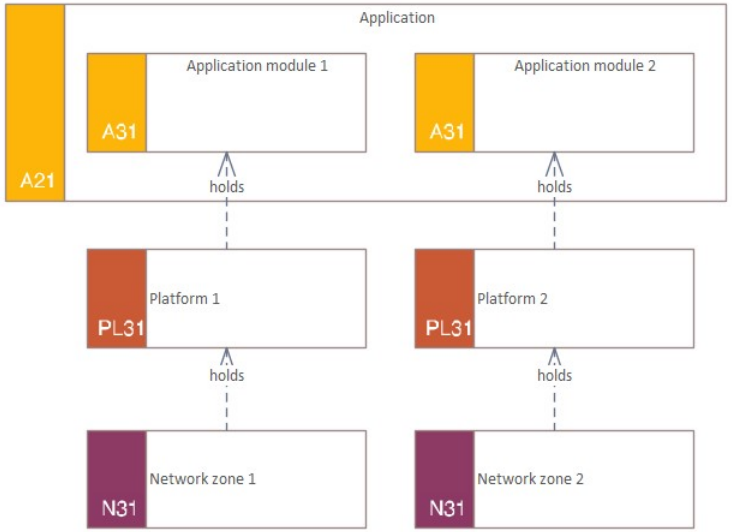
Other material that increases understanding of the needs, concerns and context

1) AB, Application description 2) IK, Integration map 3) LAM, Logical application model 4) FM, Capability model 5) IFB, Information flow description 6) AOB, Application interaction description with focus on information objects 7) AOB, Application interaction description with focus on non-functional requirements 8) Optional maps, descriptions or models

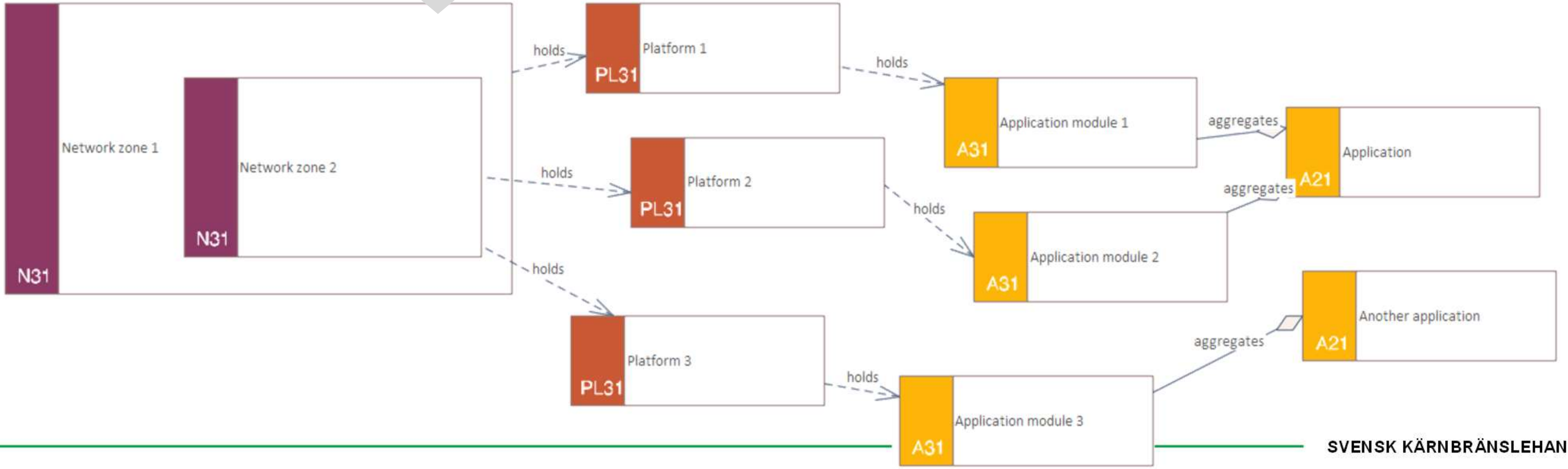
Architecture products and libraries

The deployment view ”system thinking” effect on ”enterprise thinking”

”System thinking” modelling the deployment view

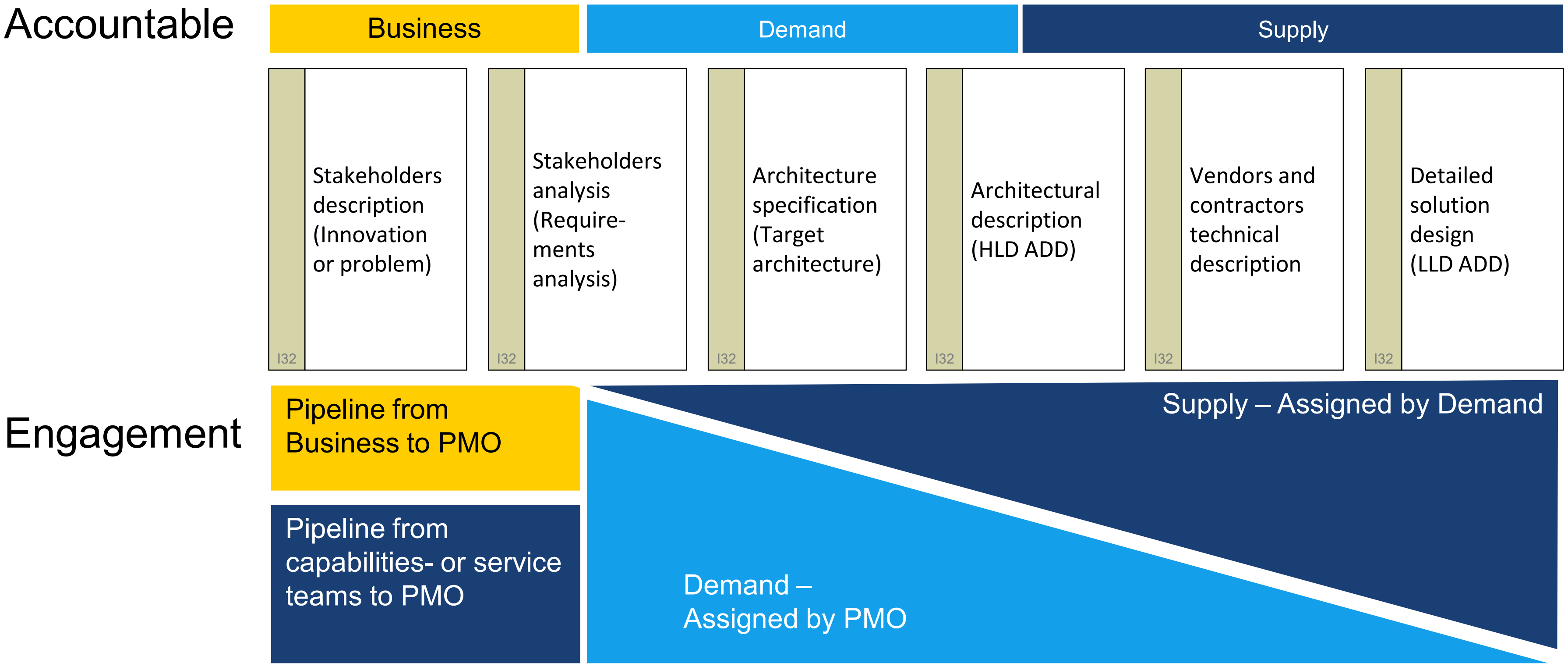


”Enterprise thinking” analyzing damage domains via the in-depth defence and its relations (zone model example)



Architecture products and libraries

Text based architectural artefacts with Sparx EA content



Accountable

Business

Demand

Supply

Stakeholders description (Innovation or problem)

I32

Stakeholders analysis (Require-ments analysis)

I32

Architecture specification (Target architecture)

I32

Architectural description (HLD ADD)

I32

Vendors and contractors technical description

I32

Detailed solution design (LLD ADD)

I32

Engagement

Pipeline from Business to PMO

Pipeline from capabilities- or service teams to PMO

Supply – Assigned by Demand

Demand – Assigned by PMO

Requirements

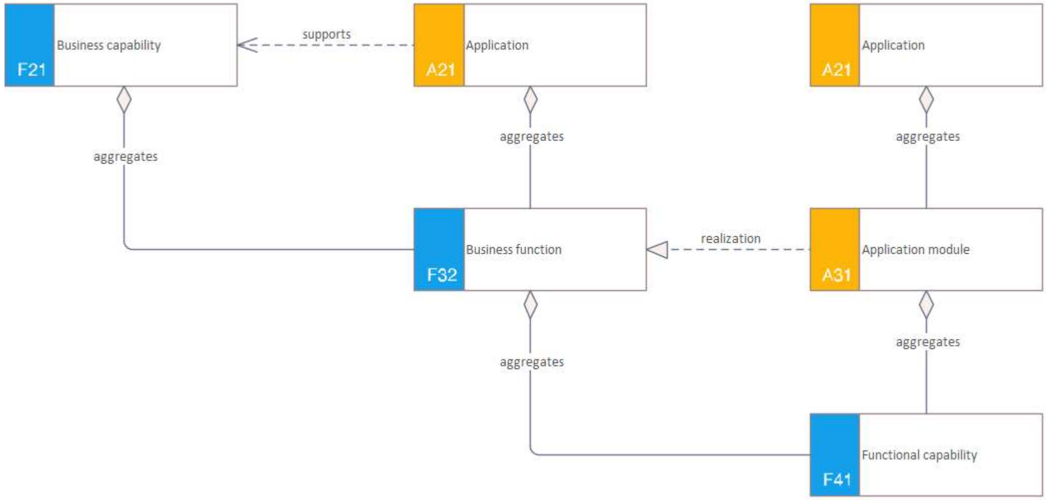
Overview

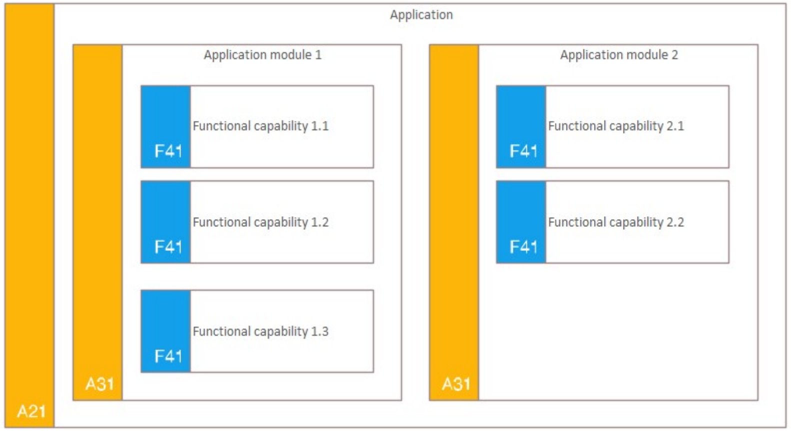
- Traceability from stakeholders demand to supply
  - Extract Business requirements to modeling
    - Integration DOORS/Excel to Sparx EA
  - Extract Security requirements to modeling
- Support in functional procurement (RFI/RFQ)
  - Export to Excel RFI/RFQ (Hi-Level or Low-Level)
  - Import from Contractors/Vendors RFI/RFQ (Excel)
- Evaluation of different solutions against requirements
  - Functional and non-functional requirements of logic applications
  - The supplier described their respective physical applications
- Gap analysis that facilitates transformation planning

68% of IS/IT-projects fails according to research

Requirements

Principles mapping capability to applications (functional requirements)





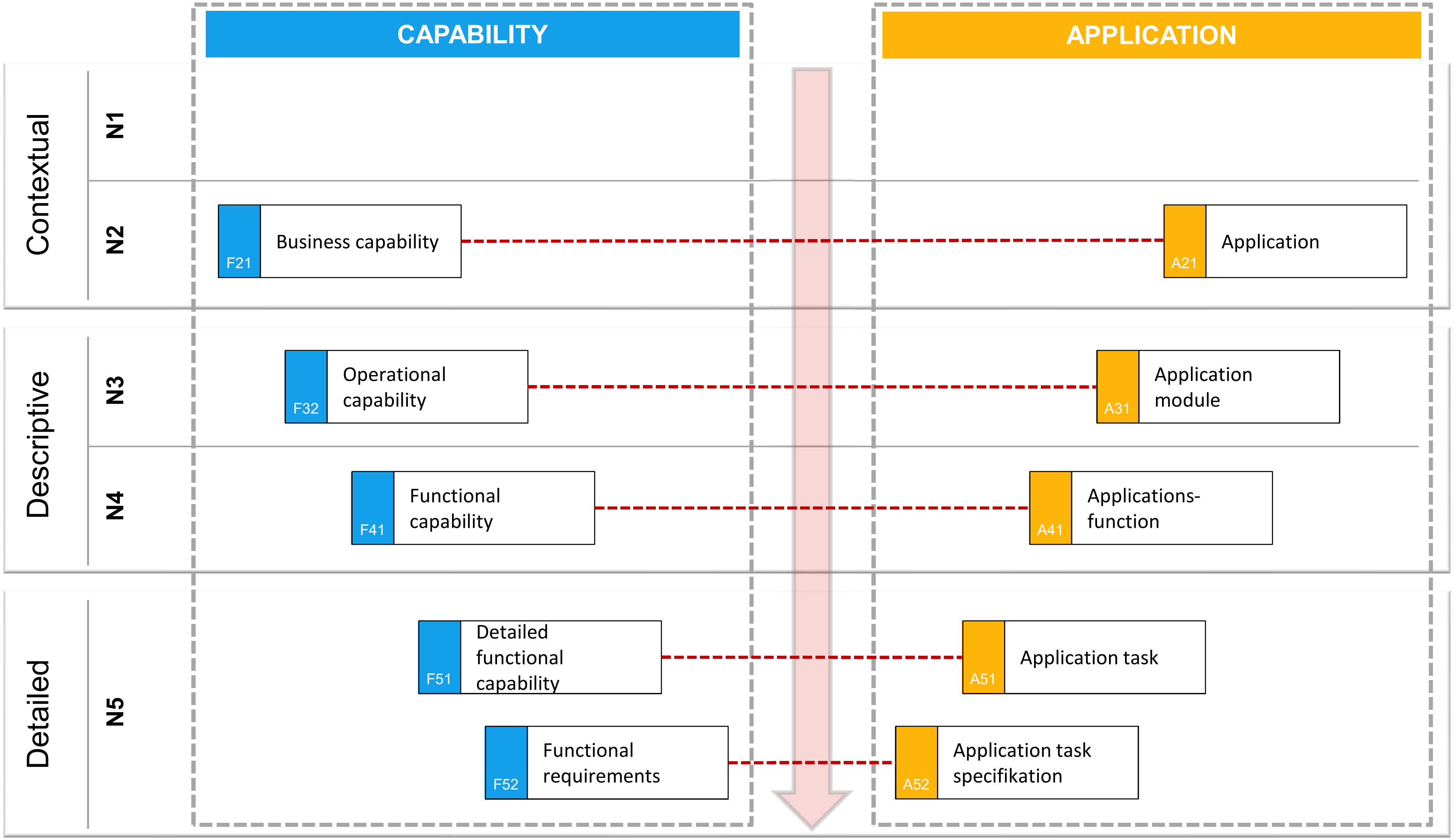
Application
Application module 1
Application module 2
F41
Functional capability 1.1
F41
Functional capability 1.2
F41
Functional capability 1.3
F41
Functional capability 2.1
F41
Functional capability 2.2
A31
A31
A21

Defining the "To be"

- Model capabilities with applications
- Match capabilities to different applications
- Analyze gaps between the capabilities wanted and existing functional blocks
- Use the capability to map capability flows or processes visualizing workflows
- Use the capability to map organizations visualizing expected roles or physical parts (assets)
- Export the capability objects as a list of functional requirements

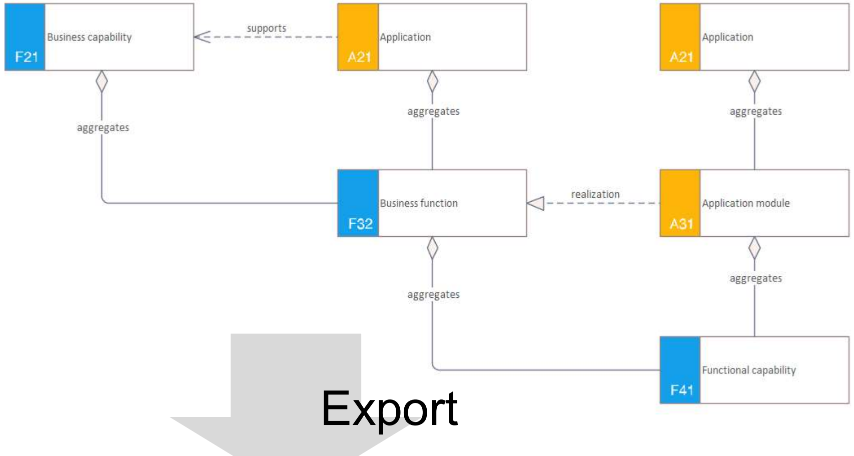
Requirements

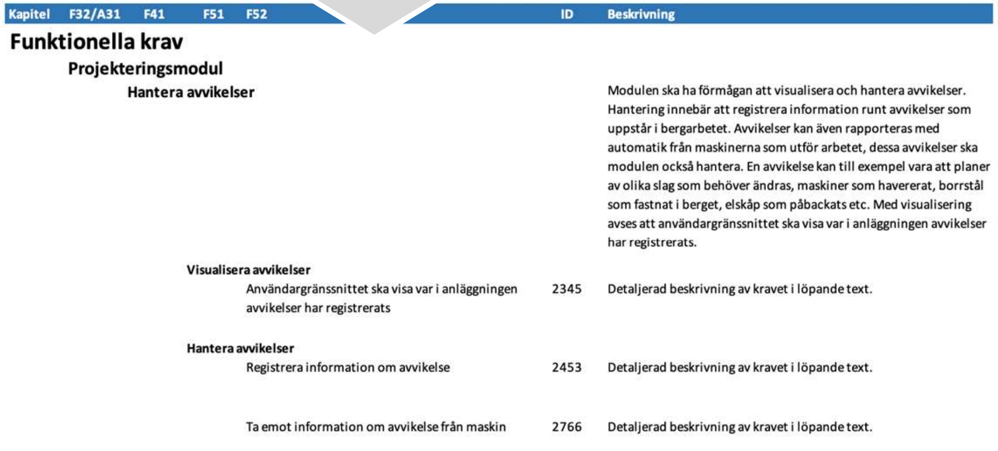
Principles for functional evaluation



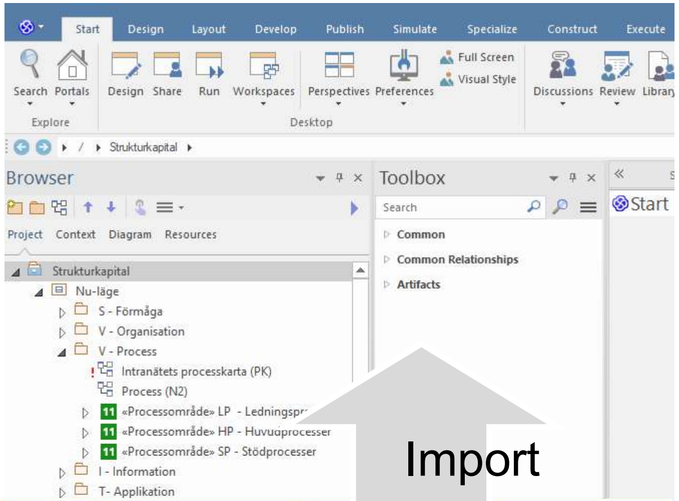
Requirements

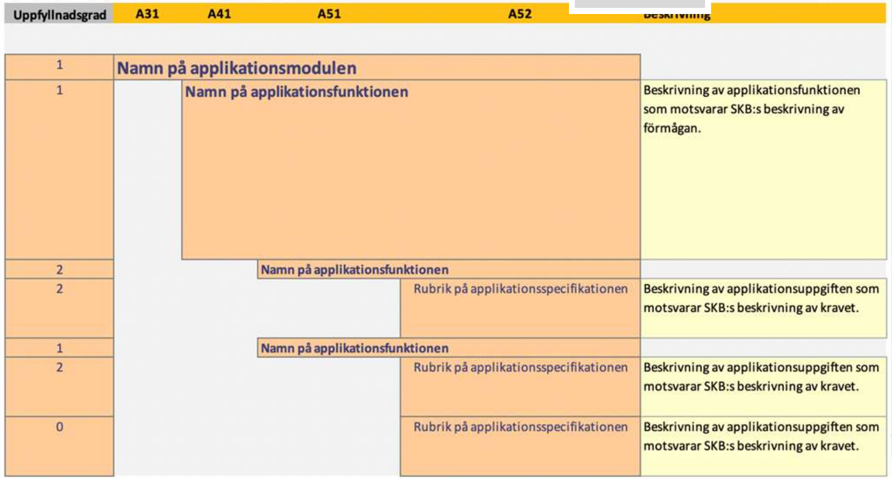
Modeling of requirements and response template





Kapitel	F32/A31	F41	F51	F52	ID	Beskrivning
Funktionella krav						
	Projekteringsmodul					
		Hantera avvikelser				Modulen ska ha förmågan att visualisera och hantera avvikelser. Hantering innebär att registrera information runt avvikelser som uppstår i bergarbetet. Avvikelser kan även rapporteras med automatik från maskinerna som utför arbetet, dessa avvikelser ska modulen också hantera. En avvikelse kan till exempel vara att planer av olika slag som behöver ändras, maskiner som havererat, borrstål som fastnat i berget, elskåp som påbackats etc. Med visualisering avses att användargränssnittet ska visa var i anläggningen avvikelser har registrerats.
			Visualisera avvikelser			
				Användargränssnittet ska visa var i anläggningen avvikelser har registrerats	2345	Detaljerad beskrivning av kravet i löpande text.
			Hantera avvikelser			
				Registrera information om avvikelse	2453	Detaljerad beskrivning av kravet i löpande text.
				Ta emot information om avvikelse från maskin	2766	Detaljerad beskrivning av kravet i löpande text.



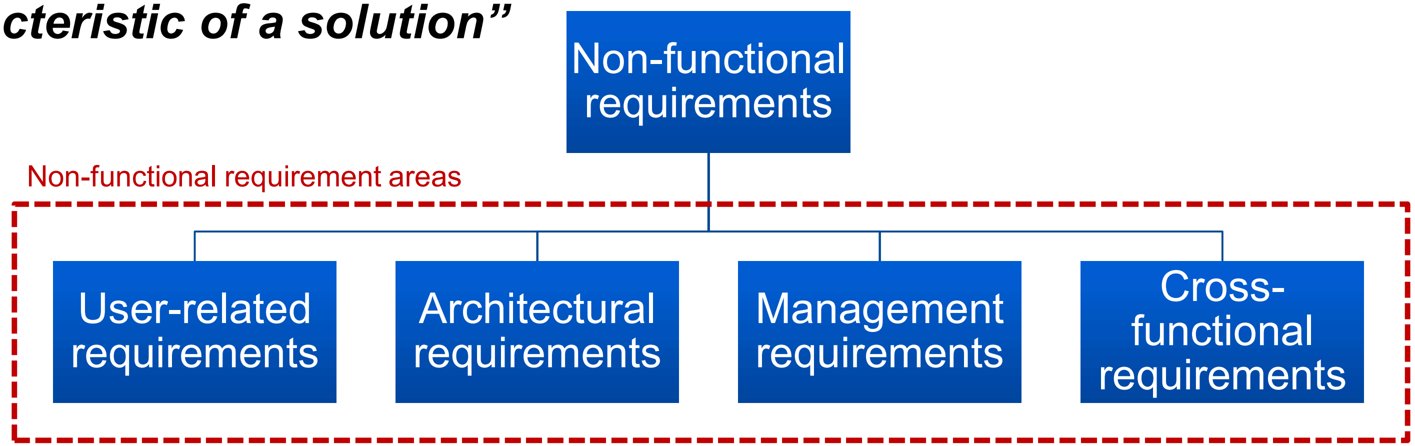


Uppfyllnadsgrad	A31	A41	A51	A52	Beskrivning
1	**Namn på applikationsmodulen**				
1		**Namn på applikationsfunktionen**			Beskrivning av applikationsfunktionen som motsvarar SKB:s beskrivning av förmågan.
2			**Namn på applikationsfunktionen**		
2				Rubrik på applikationsspecifikationen	Beskrivning av applikationsuppgiften som motsvarar SKB:s beskrivning av kravet.
1			**Namn på applikationsfunktionen**		
2				Rubrik på applikationsspecifikationen	Beskrivning av applikationsuppgiften som motsvarar SKB:s beskrivning av kravet.
0				Rubrik på applikationsspecifikationen	Beskrivning av applikationsuppgiften som motsvarar SKB:s beskrivning av kravet.

Requirements

Non-functional requirements, areas

"A statement about a requested quality characteristic of a solution"



Non-functional requirements

Non-functional requirement areas

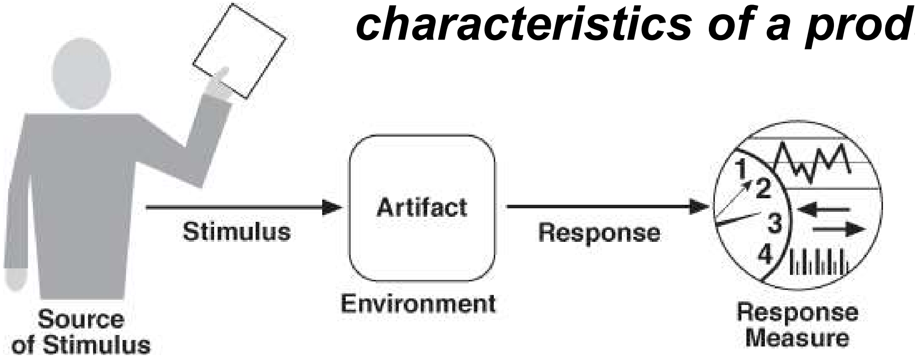
User-related requirements

Architectural requirements

Management requirements

Cross-functional requirements

"Quality attributes are characteristics of a product"

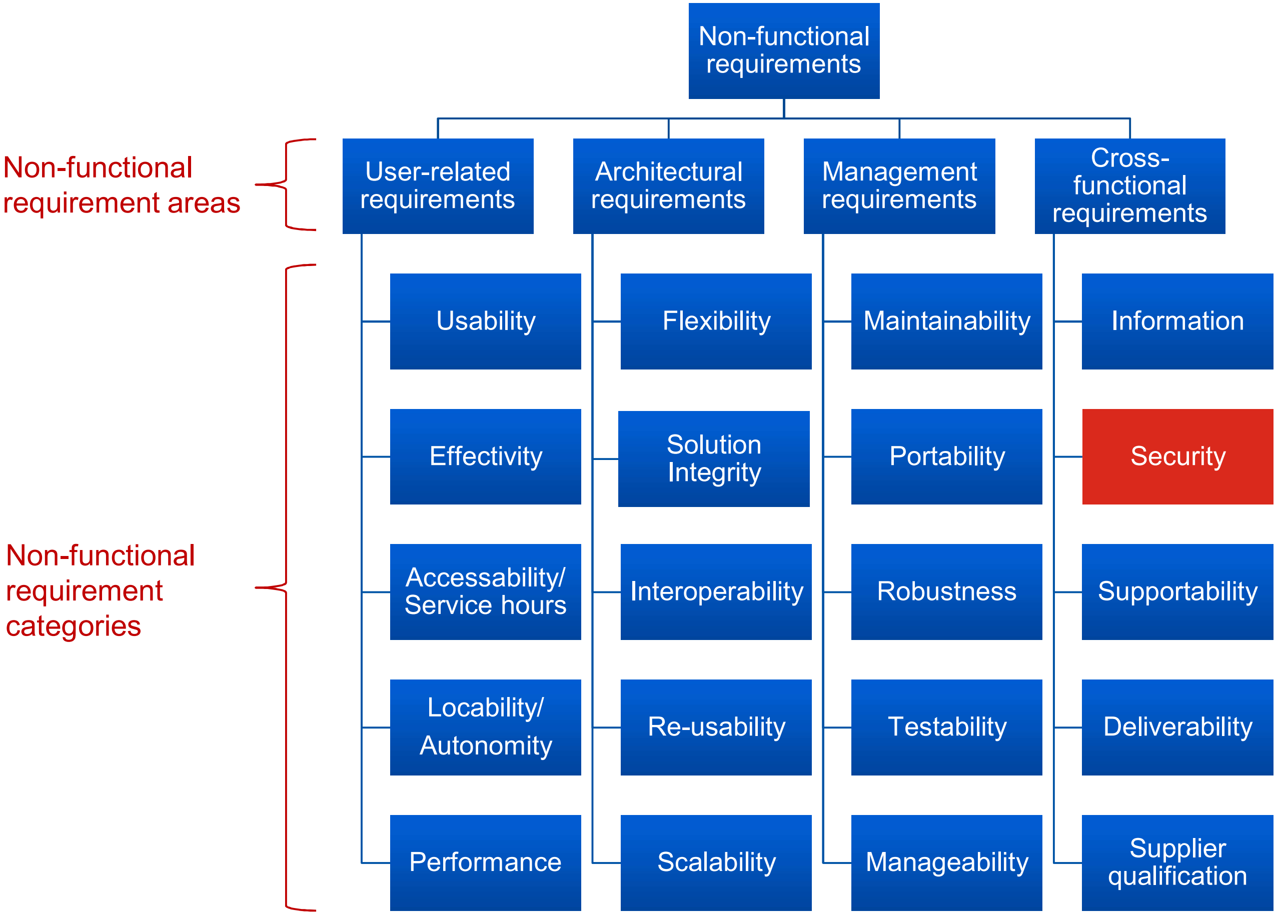


"Non-functional requirement is the measure and criteria of such characteristics"

Quality attributes for service-oriented architectures - L O'Brien, P Merson, L Bass

Requirements

Non-functional requirements, categories



Non-functional requirements

Non-functional requirement areas

Non-functional requirement categories

User-related requirements	Architectural requirements	Management requirements	Cross-functional requirements
Usability	Flexibility	Maintainability	Information
Effectivity	Solution Integrity	Portability	Security
Accessability/ Service hours	Interoperability	Robustness	Supportability
Locability/ Autonomity	Re-usability	Testability	Deliverability
Performance	Scalability	Manageability	Supplier qualification

ISO 27000

- Cofidentiality
- Integrity
- Aviability
- Traceability

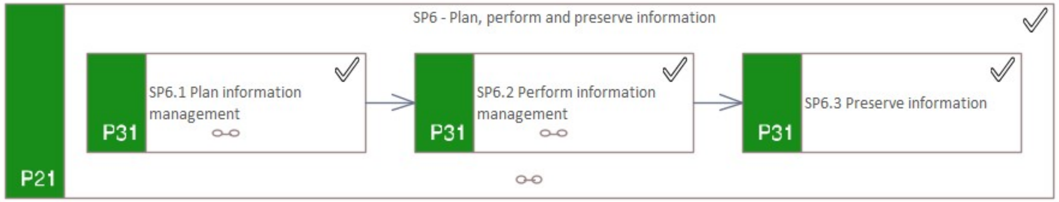
…and regulatories

Lets explore modelling of non-functional requirements in the security chapter…

Security

Overview

- Classification of information (ISO 27001)
  - Type defined information management SOP:s Plan, Perform, Preserve
  - Confidentiality, Accuracy, Availability, Traceability
  - GDPR and other regulatory classifications
- The security-dimensions (non-functional requirements and risks)
- “Dynamic” security requirements library



SP6 - Plan, perform and preserve information

SP6.1 Plan information management

SP6.2 Perform information management

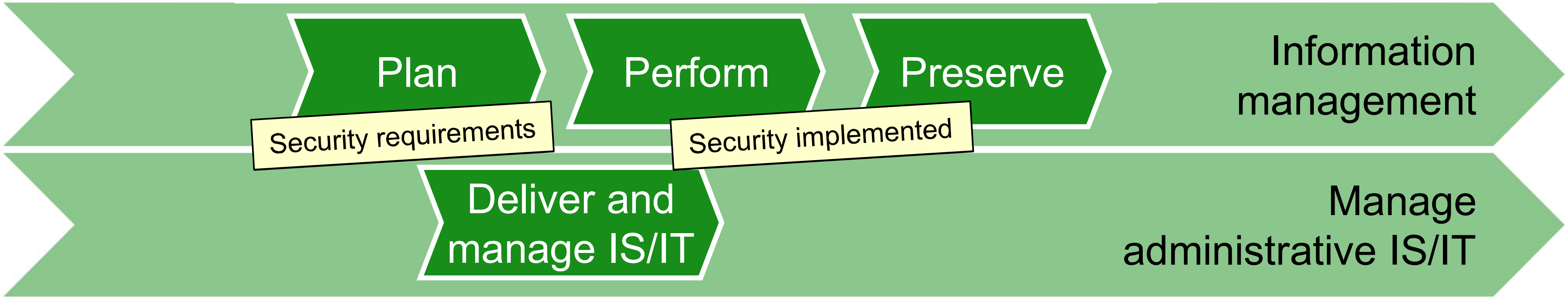
SP6.3 Preserve information

P31

P31

P31

P21



Plan

Perform

Preserve

Information management

Security requirements

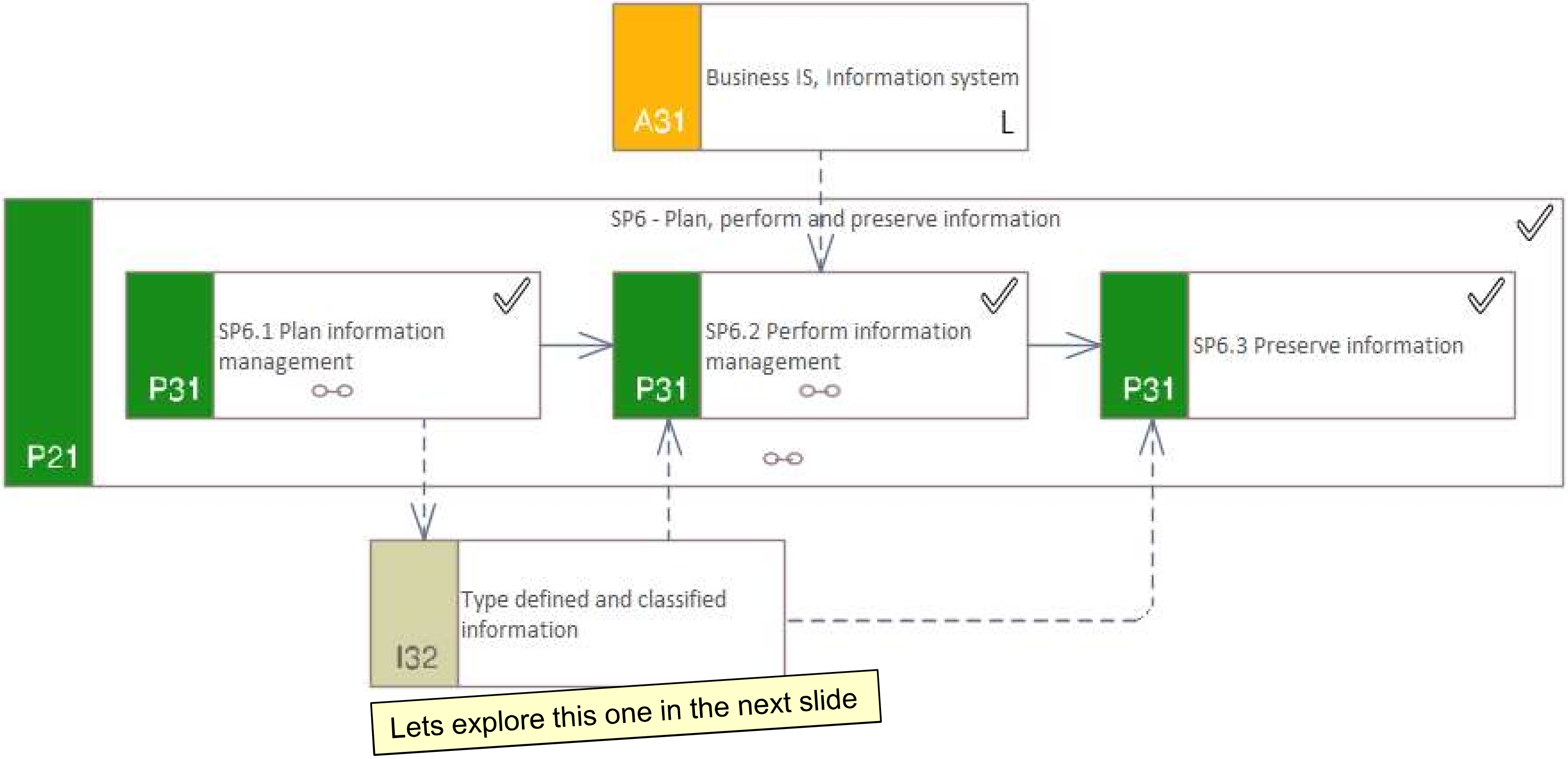
Security implemented

Deliver and manage IS/IT

Manage administrative IS/IT

Security

Method of finding non-functional requirements and make them functional


A31

Business IS, Information system

L

P21

SP6 - Plan, perform and preserve information ✓

P31

SP6.1 Plan information management ✓

P31

SP6.2 Perform information management ✓

P31

SP6.3 Preserve information ✓

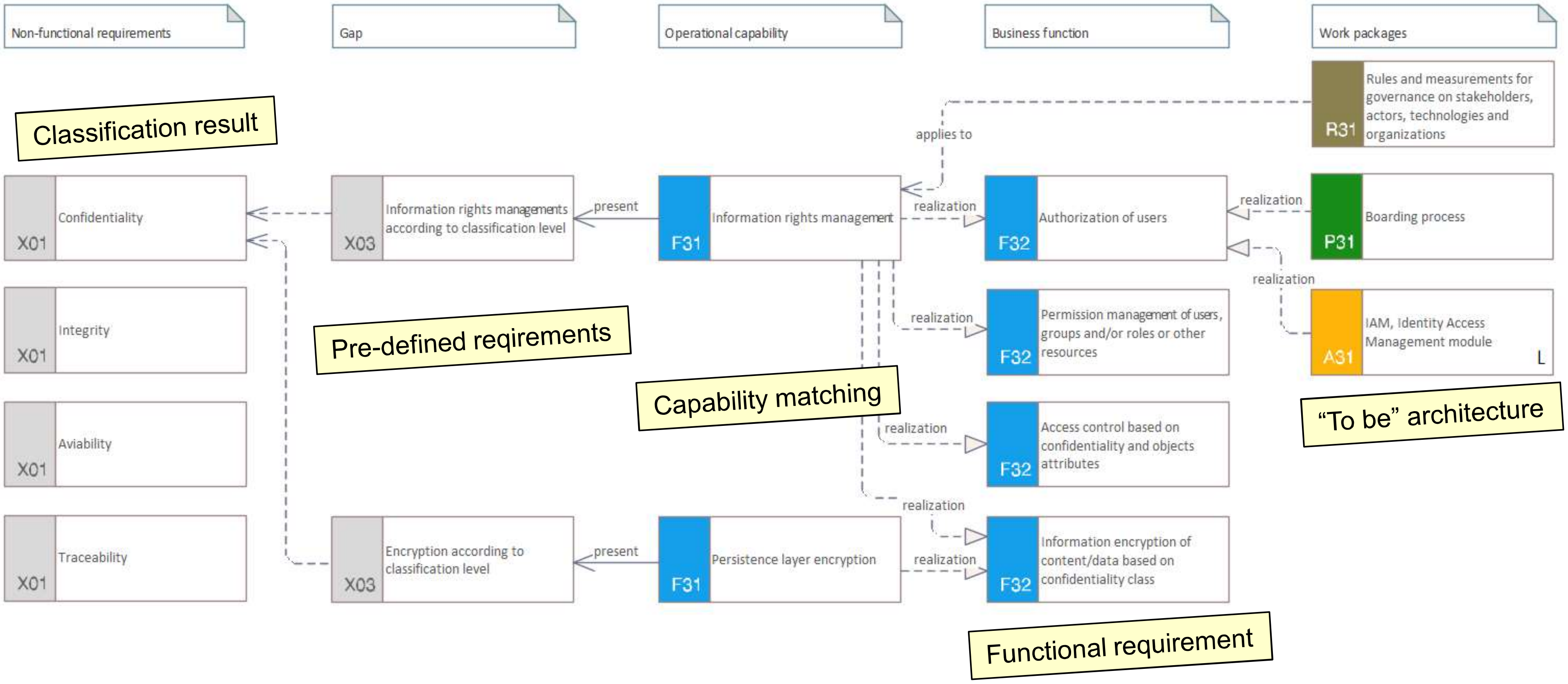
I32

Type defined and classified information

Lets explore this one in the next slide

Security

Modelling non-functional requirements



Non-functional requirements

Gap

Operational capability

Business function

Work packages

Classification result

X01 Confidentiality

X01 Integrity

X01 Aviability

X01 Traceability

Pre-defined reqirements

X03 Information rights managements according to classification level

X03 Encryption according to classification level

present

present

Capability matching

F31 Information rights management

F31 Persistence layer encryption

applies to

realization

realization

realization

realization

realization

F32 Authorization of users

F32 Permission management of users, groups and/or roles or other resources

F32 Access control based on confidentiality and objects attributes

F32 Information encryption of content/data based on confidentiality class

Functional requirement

realization

realization

R31 Rules and measurements for governance on stakeholders, actors, technologies and organizations

P31 Boarding process

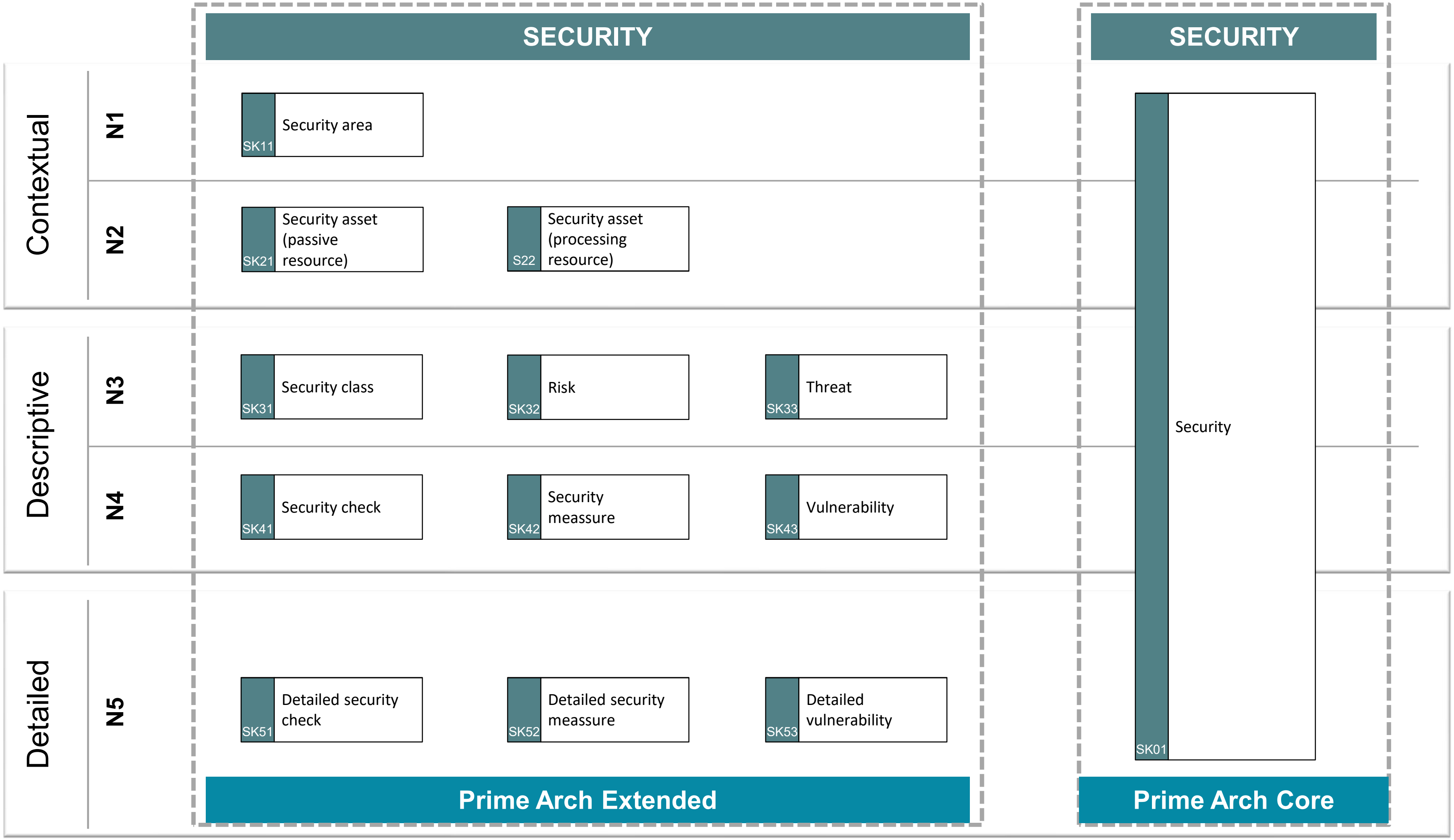
A31 IAM, Identity Access Management module

L

“To be” architecture

Security

“The upcoming scope” in the Prime Arch Framework – Security dimension



		SECURITY			SECURITY
Contextual	N1	SK11 Security area			SK01 Security
	N2	SK21 Security asset (passive resource)	S22 Security asset (processing resource)		
Descriptive	N3	SK31 Security class	SK32 Risk	SK33 Threat	
	N4	SK41 Security check	SK42 Security meassure	SK43 Vulnerability	
Detailed	N5	SK51 Detailed security check	SK52 Detailed security meassure	SK53 Detailed vulnerability	
		Prime Arch Extended			Prime Arch Core