

# **Architecting digital repeatable systems** for systemic Digital Transformation

Module 1-2

System, digital and architecture

Dr Alexander Samarin

**SAMARIN.BIZ**



# Definition of system (1)

- **object**  
anything perceivable or conceivable
  - Note 1: Objects may be
    - material (e.g. an engine, a sheet of paper, a diamond)
    - immaterial (e.g. a conversion ratio, a project plan)
    - imagined (e.g. a unicorn)
- **entity**  
*object* with an identity

# Definition of system (2)

- **system**

representation of an entity as a complex of interacting entities organised as a whole which exhibits (as the result of interaction between the elements) some emergent characteristics indispensable to achieve one or more stated purposes

- Etymology: from the Latin *systema*, which in turn is derived from the Greek *σύστημα* meaning “together”

- A system is more than a sum of its entities

- Those entities are called also “discrete elements”

- “Discrete” means “individually separate and distinct”



# System's characteristics

- A system is encapsulated, has a **boundary**
  - A system can be nested inside another system
  - A system can overlap with another system
- A system is bounded in **time**; it has its own life cycle
- A system is bounded in **space**, though the elements are not necessarily co-located
- A system receives input from, and sends output into, the wider **environment**
- A system consists of **processes** majority of them transform inputs into outputs
- Different stakeholders see the same system **differently**
- A system for **somebody** is an element for another

# Related concepts

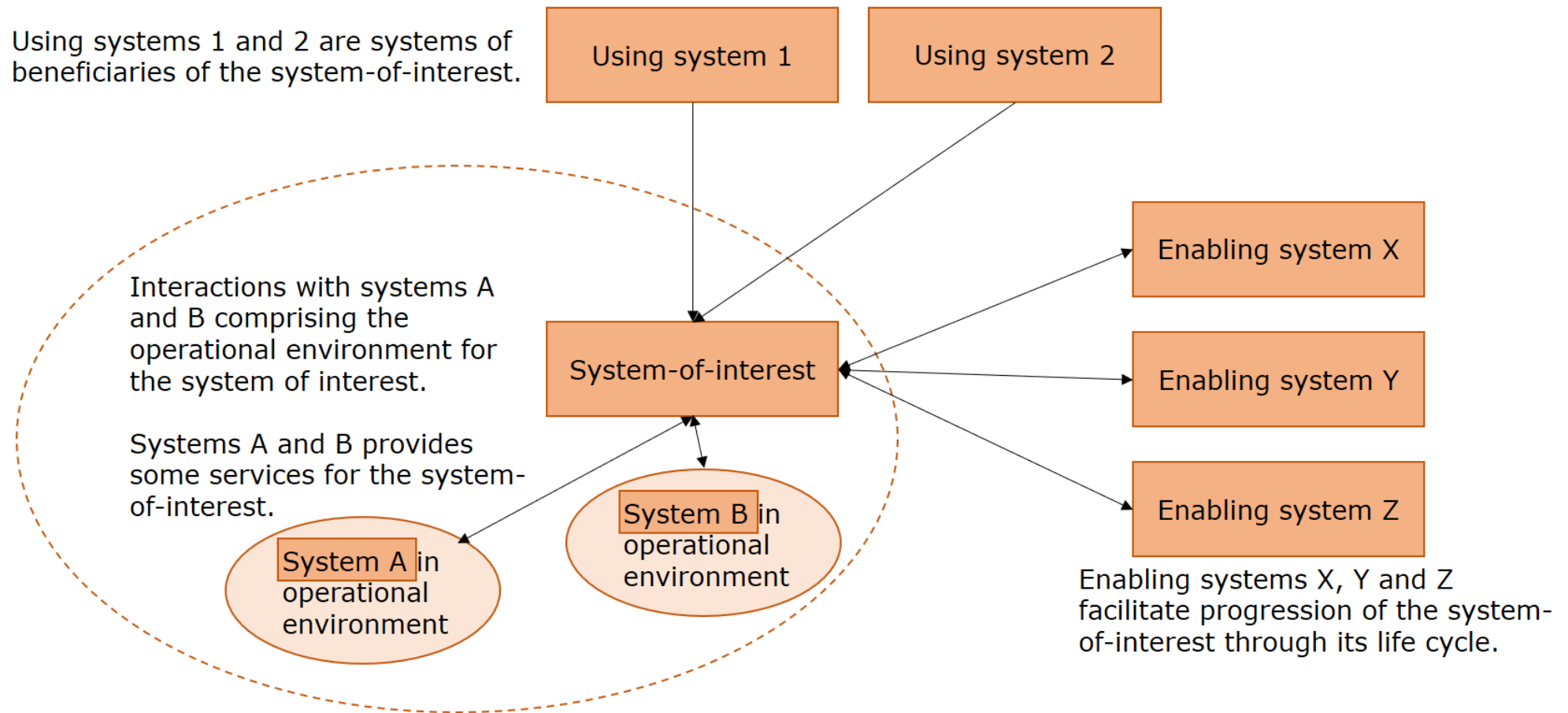
- **structure**, <of a system>  
internal arrangement of, and relationship between, elements
  - Etymology: from the Latin *structūra* meaning “a fitting together”
- **behaviour**, <of a system>  
what a system will do in response to its external environment without referring to details on implementation
  - In some cases, the behaviour of a system cannot be explained in terms of the behaviour of its elements
  - Some emergent characteristics are the result of a system’s behaviour
- **stakeholder**, <of a system>
  - person, group of persons or organization having an interest in a *system-of-interest*

# Any system has a structure

- Chaotic vs. organised
- Accidental vs. intentional
- Static vs. dynamic



# System-of-interest and its environment





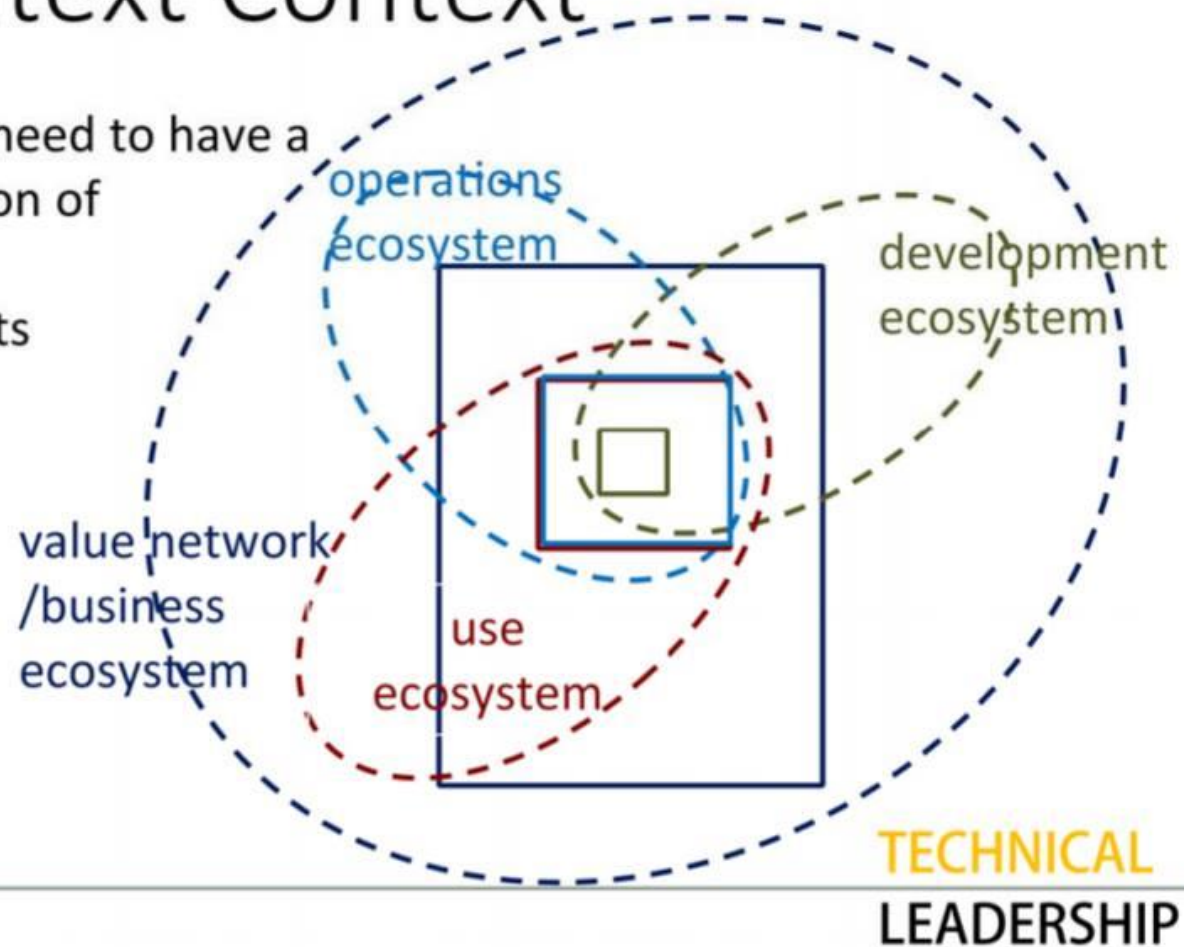
# Context Context Context

To make a decision, we need to have a (good enough) conception of

- Desired outcome(s)
- Forces and constraints

Arising in context of

- development
- operations
- use
- value network





# Systems are complex

- Unlimited life-cycle (unpredictable and incremental evolution)
- Socio-technical
- Collaborative
- Industrialised
- Ability for rapid innovation is important
- Variety of services
- High level of security for data
- Customer experience
- Self-referential

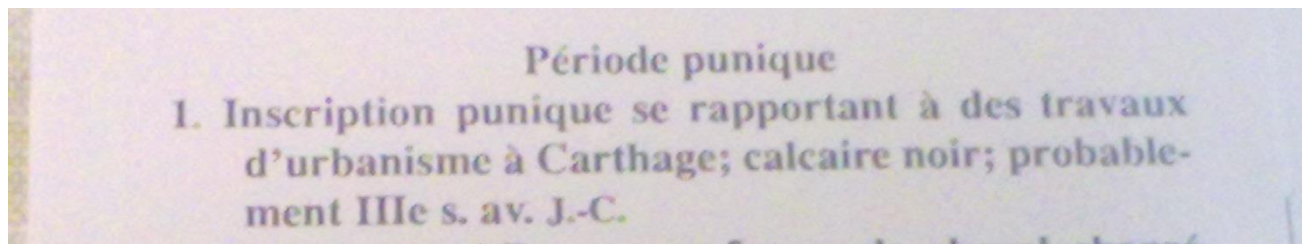
# Systems architecting

- **architecture** <of a system>  
fundamental orderliness (embodied in its components, their relationships to each other and the environment), and the principles governing the design, implementation and evolution
- **architecting** is about:
  - making essential decisions about the system-in-focus to enable the achievement of its desired emergent characteristics
  - understanding the relationship between structure and behaviour, between design and outcomes
- An **architect** is a person who
  - a) translates a customer's requirements into a viable plan and
  - b) guides others in its execution



# Architecture used to construct Carthage

- Enterprise architecture is sometimes translated into French as “urbanisation”





# Importance of architecture

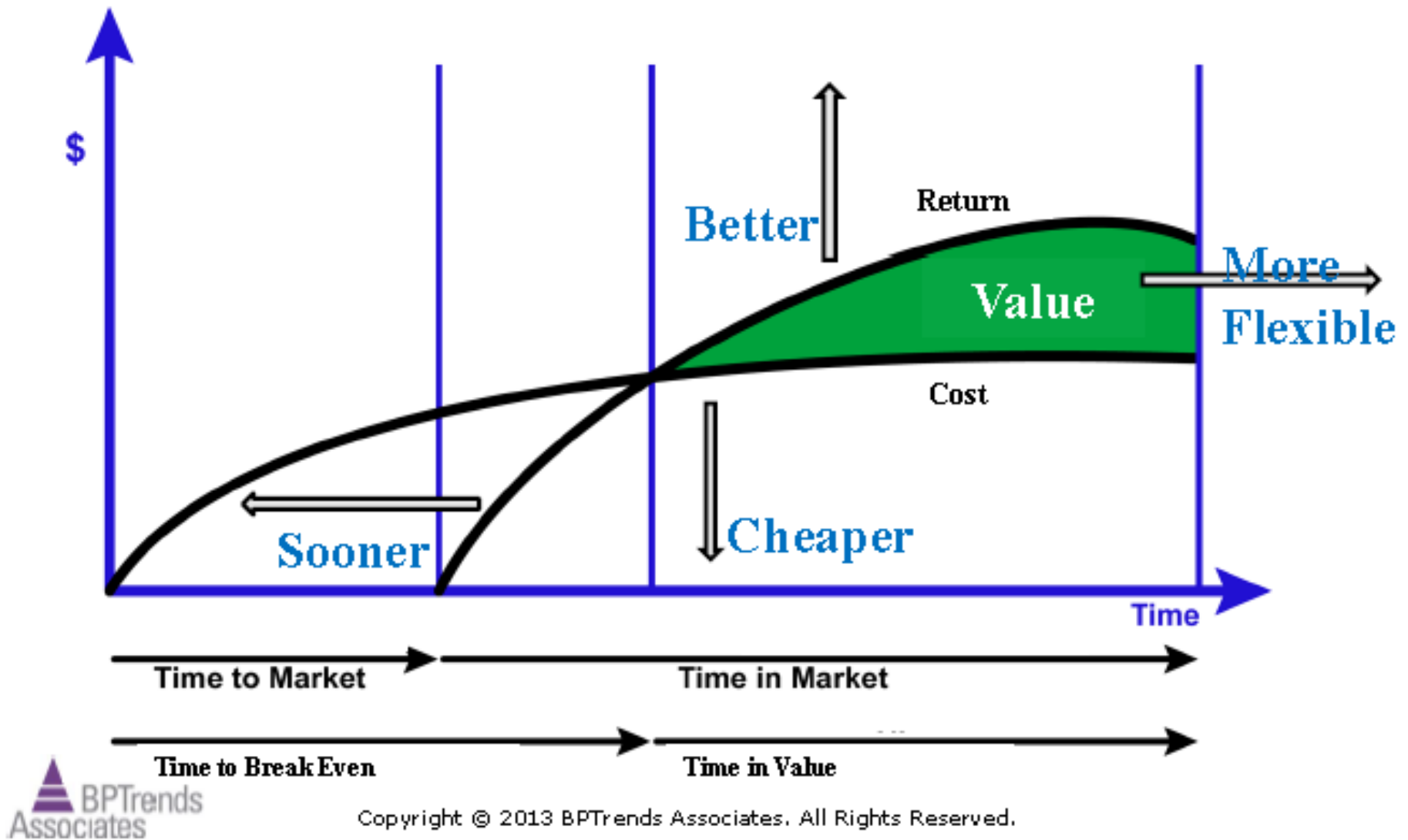


- No architectural blueprint
- 38 years of construction
- 160 rooms, 497 doorways, 950 doors
- Over 20 tonnes of paint required



- No disruption of river traffic activities
- The committee evaluated 50 projects
- Three architectural techniques
- 8 years of construction
- Modernised for new technology

# The value of business innovation



# EA definition

- **Enterprise Architecture (EA)** is a system-thinking applied management discipline for coordinating people, processes, projects and products
  - **System-thinking applied management discipline** is an applied management discipline which uses system-thinking approach
  - **Applied management discipline** is a management discipline which applies scientific knowledge for solving practical problems
  - **Management discipline** is a discipline for the better management of the enterprise functioning in support of the enterprise goals
  - **Discipline** is a coherent set of governing rules

# Managing complexity

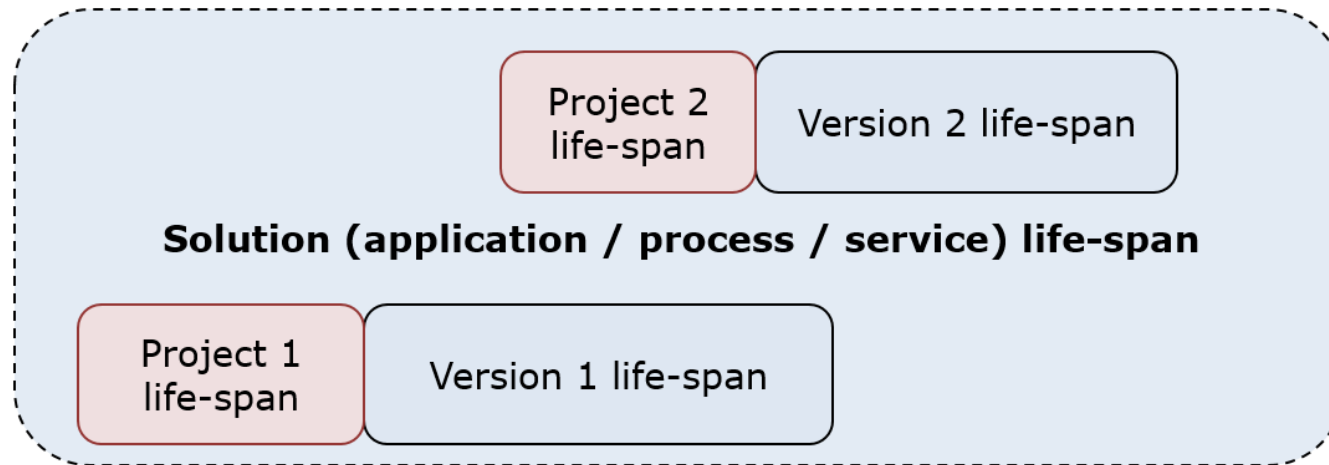
- Three different sources of complexity:
  - **natural** complexity (problem space)
  - **cultural** complexity (social space)
  - **undesired** complexity (solution space)
- The purpose of Enterprise Architecture (EA)
  - promote the use explicit and executable techniques to reduce the natural complexity
  - guide solution architecture to follow the natural complexity to avoid adding undesired complexity
  - liberate resources to handle the natural complexity



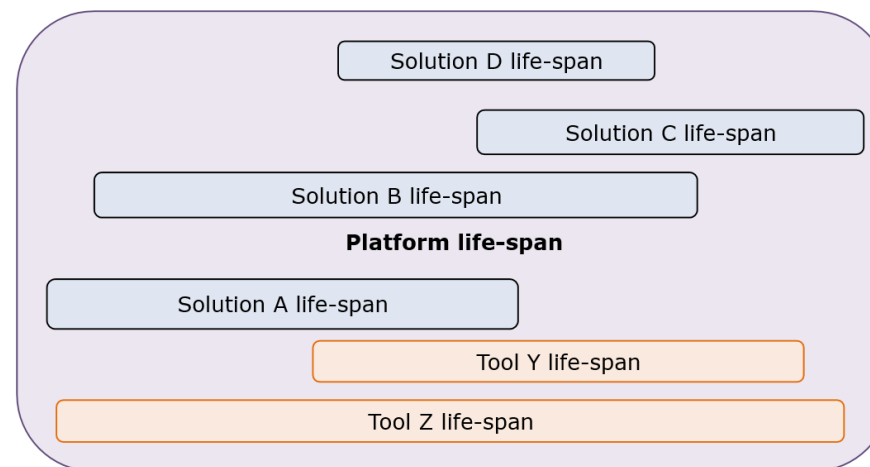
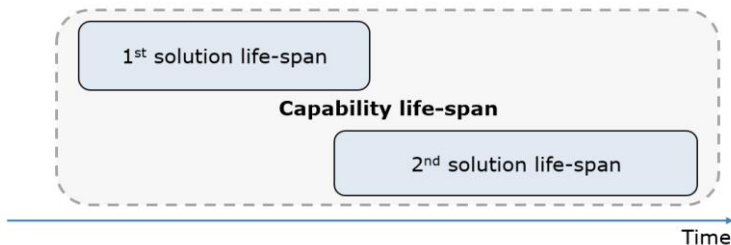
# EA coordination dimensions

- 1. interest space** (or scope)
- 2. architectural space** (as TOGAF layers)
- 3. time span** (technology life-cycle, enterprise life-cycle)
- 4. sector span** (communalities between various sectors)
- 5. problem space** (digitalisation, re-structuring, rationalisation, M&A, modernisation, etc.)
- 6. solution space** (ZF rows: scope, concept, logical, technical, implementation, operations)
- 7. cultural space**
- 8. practice space** (theoretician methodologist practitioner)
- 9. media space** (physical vs digital)
- 10. financial space** (number of zero in the budget)
- 11. people space** (top, management, middle-management, super-users, workers)

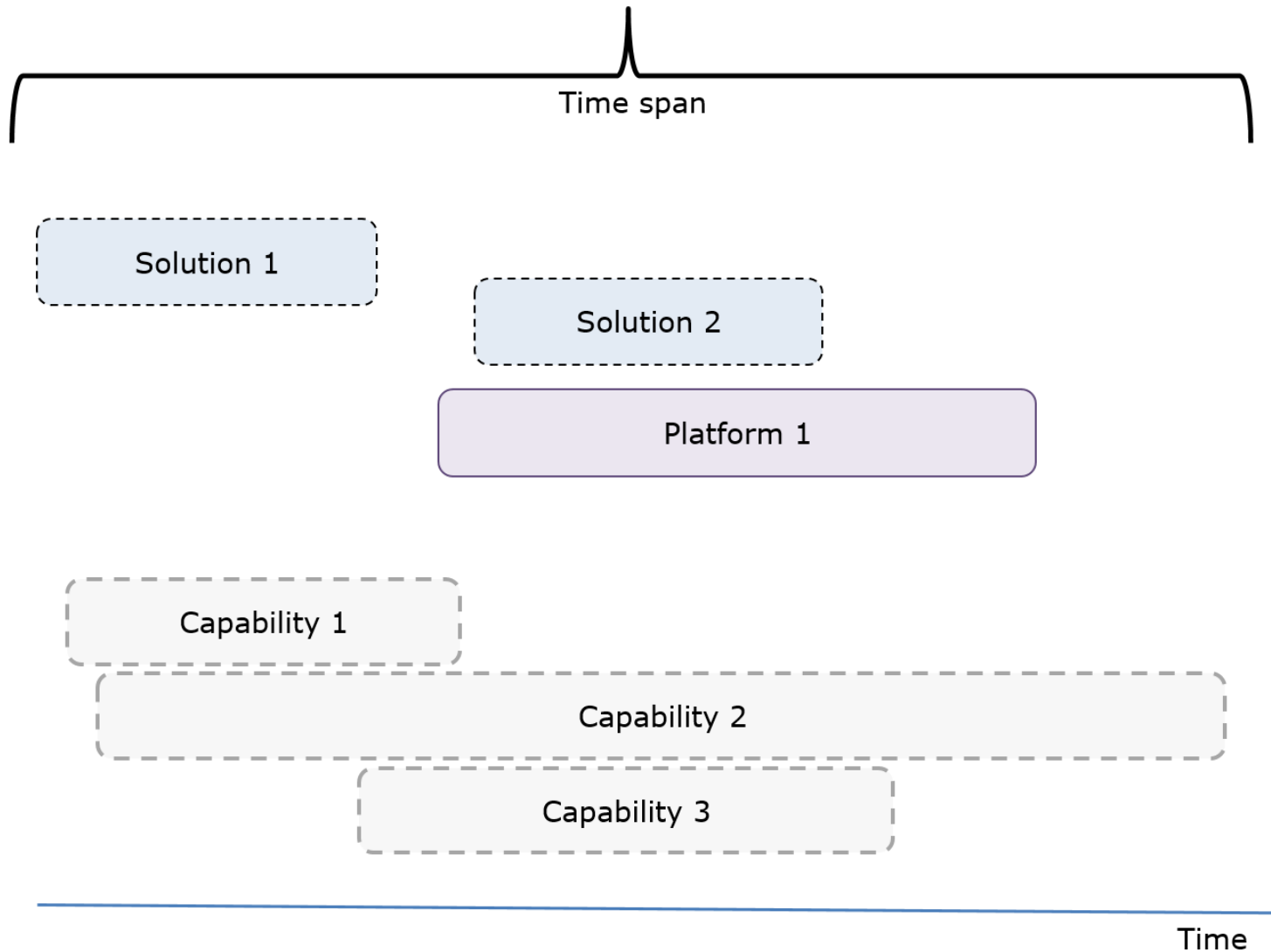
# EA viewpoints: projects, solutions, capabilities and platforms



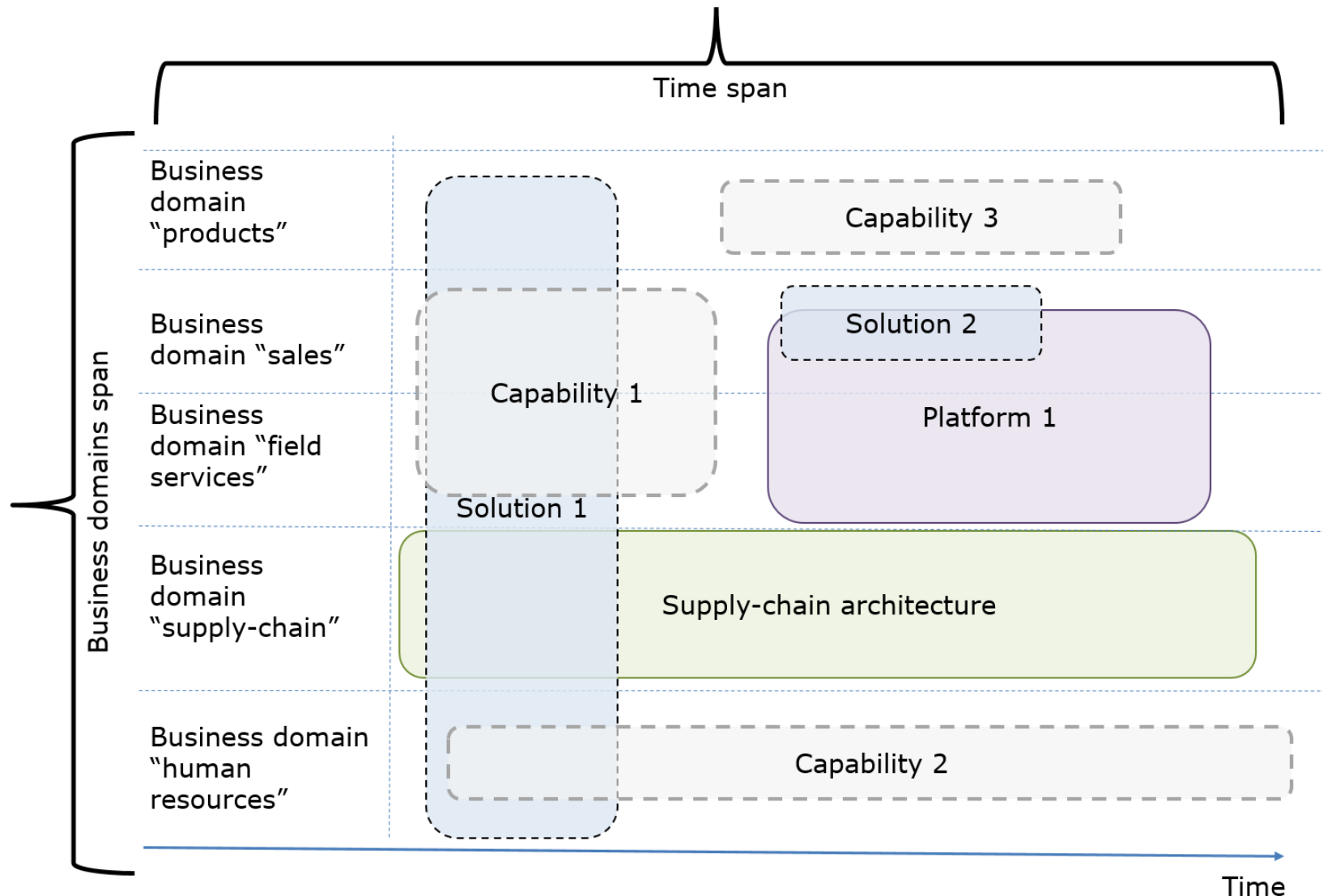
Time



# EA viewpoints: time span



# EA viewpoints: business domains span vs time span



# EA viewpoints: architectural practices span vs business domains span

		Business domains span				
		"human resources"	"supply-chain"	"field services"	"sales"	"products"
Architectural practices span	Application architecture	++	+++	+	+	+
	Business architecture	+	+	++	+	+++
	Data architecture	+	+++		+++	
	Information architecture		++		++	++
	Infrastructure architecture	+	++	+	+	
	Process architecture	+	+++	++		+
	Security architecture		+	+		+
	Technology architecture	+	+	++	+	

# Enterprise architect qualification

- interest space
  - work for 1000+ people; able to talk to everyone
- architectural space
  - able to discuss all domains; expert in 2-3 domains
- time span
  - full enterprise life-cycle; lead important changes
- sector span
  - 5 different sectors; able to find similarities between sectors
- problem space
  - leading min 3 critical enterprise-wide changes
- solution space
  - comfortable in min 4 rows
- cultural space
  - know culture specifics for the majority of staff members
- practice space
  - be good at least in 1 of these roles
- media space
  - make fully digital at least one enterprise
- financial space
  - manage projects with the budget of 6+ figure
- people space
  - be able to explain to everyone how EA will address his/her concerns

# **Only EA can systematically solve enterprise-wide problems**

- capability benchmarking
- QMS as an initial business architecture
- compliance
- decentralization
- ERM
- restructuring
- strategy execution
- info security
- application modernisation
- paperless office
- shared services
- e-government, smart-city, e-health
- digital transformation



# Various approaches (1)

- System approach
  - holistically, hierarchically, structured, multiplicity of models
- Functional (component) approach
  - each component is optimised separately
- Situational approach
  - all depends on a particular situation
- Process approach
  - any functioning is a process
- Project approach
  - applying process approach to unique operations

# Various approaches (2)

- Architectural approach
  - **decomposes** a system into its component parts (analysis)
  - formalizes the description of the **relationships** between components (model)
  - finds **opportunities** for structural and behavioural good practices (patterns)
  - uses components and relationships to **define** other components (synthesis)
  - optimizes performance (simulation)

# Main architecting principles

- Principles
  - all **artefacts** (types of component) are versionable throughout their life-cycle
  - all artefacts are evolved to become **digital**, externalised, virtual and located in clouds
  - all relationships between these artefacts are **modelled explicitly**
  - all models are made to be **executable** (actionable)
- Since many models are designed for people, these models should not be very formal but explicit
- Enable changes at the pace of the business

# Relationships between artefacts is our primary interest

- Formalise and perfect these relationships to being **executable** (actionable)
- Obtain a **full set** of relationships
- Detect the **system-forming** artefact(s)

# Systems architecting methodology

- Why do we need methodology?
  - the goal: **different** people in **similar** situations find **similar** solutions/services/components or bring innovations
- **systems approach**

holistic approach to understanding a system and its discrete elements in the context of their behaviour and their relationships to one another and to their environment

  - Note: Use of the systems approach makes explicit the structure of a system and the rules governing the behaviour and evolution of the system

# Eight areas of six systems traditions

Eight dimensions of six systems traditions (Dent & Umpleby 1998, 2005, 2012)							
	System dynamics	Total quality management	Operations research	Organizational learning	General systems theory	Cybernetics	TOTAL
From entities to relationships	Y	Y	Y	Y	Y	Y	6
From reductionism to holism	Y	Y	Y	Y	Y	Y	6
From linear to circular causality	Y	Y	Y	Y	Y	Y	6
From environment-free to environment-full investigations	Y	Y	Y	Y	Y	Y	6
From not-knowing subjects to knowing subjects	N	Y	N	Y	Y	Y	4
From determinism to indeterminism	N	N	Y	N	Y	Y	3
From not including the observer to including the observer	N	N	N	Y	N	Y	2
From direction to self-organization	N	N	N	N	N	Y	1
TOTAL	4	5	5	6	6	8	

# Digital age (1)

- Digital triumphs physical
- Fast triumphs slow
- Group triumphs single
- Big triumphs small
- With this new speed and scale, there is no time for human intervention and errors in routine operations and at interfaces



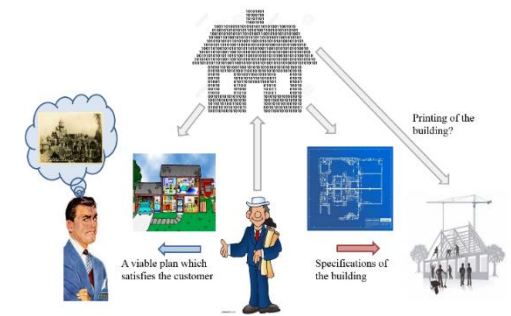


# Digital age (2)

- Transparency is increasing
- In addition to being: cheaper, faster, better
- It is mandatory to become: cleaner, greener, more agile, more synergetic (i.e. IoT)
- In systems architecting the focus is changing
  - **FROM** the thing
  - **TO** how the thing changes
  - **SUBJECT** how things change together
- To enable innovations
  - “in the digital age innovation depends on process automation”



# A digital manifesto



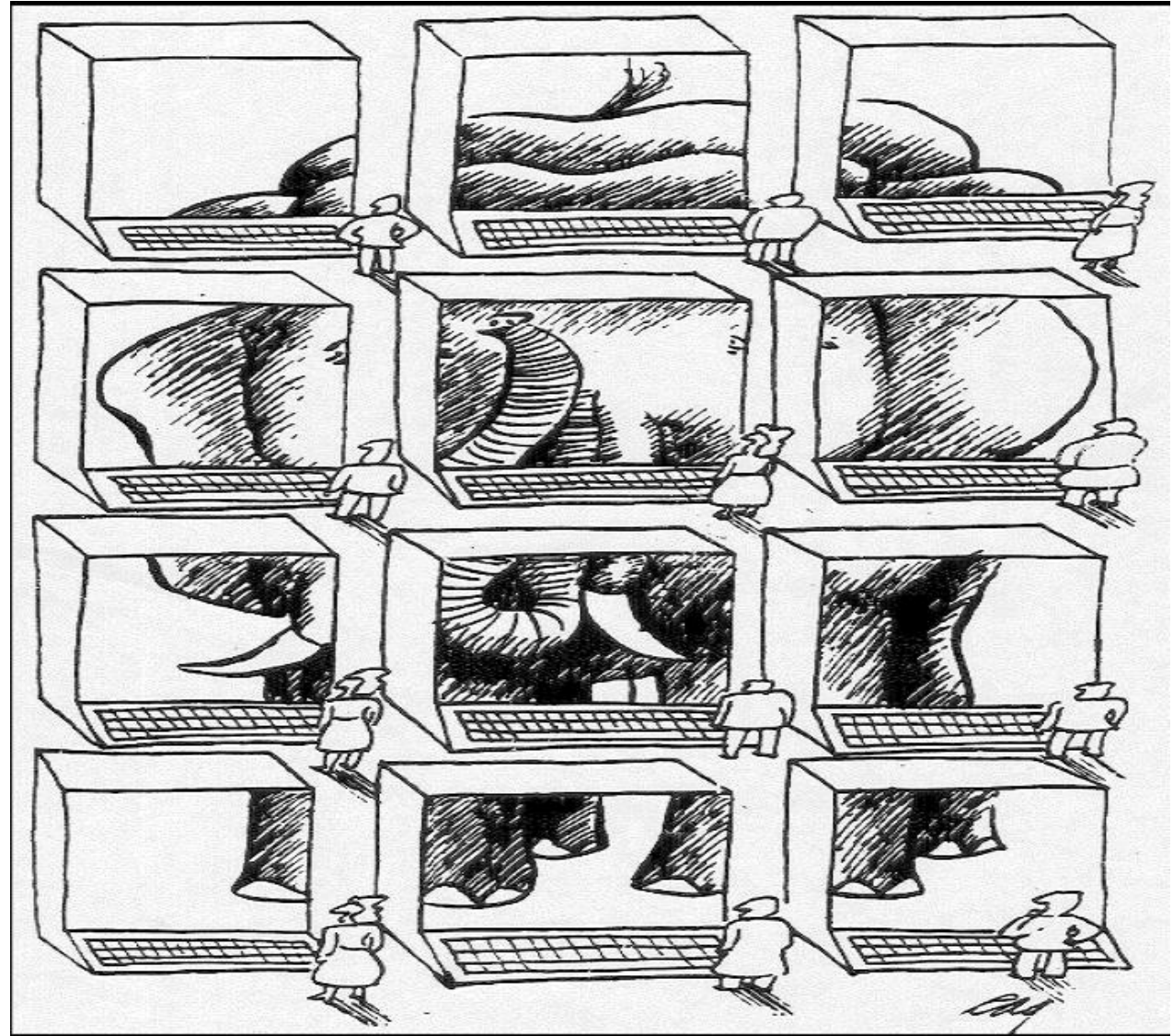
- Primary artefacts are available in **digital formats (representations, media)** (thus formal and machine-executable)
- Ideally, digital is the **master media** for primary artefacts
- Primary artefacts can be moved **between digital, analogue and physical medias** (e.g. with 3D printing and capturing techniques)
- Enterprise, ecosystem and society “**understand**” the digital formats for primary artefacts
- Enterprise can transmit, protect, validate, enrich, interpret and manipulate digital primary artefacts at their **whole life cycle**
- Enterprise knows all the **dependencies** between its digital primary artefacts
- Enterprise can generate new **knowledge** from digital primary artefacts
- Enterprise can **adapt** digital primary artefacts (extract, combine, change presentation, convert, etc.) to fit the current needs of a particular customer
- People can **delegate** to "things" (i.e. computers, sensors, actuators, robots, etc.) some routine activities with their primary artefacts (e.g. with the use of IoT)
- With the progress of IoT, "things" become **more capable actors** of digital primary processes ("things" may form temporary groups to carry out a particular activity)

# Ideal (happy) path

- Formalise your **artefacts** – syntax and semantic
- Define their **life cycles**
- All artefacts must be **versionable** throughout their lifecycle
- All artefacts must evolve to become **externalised, virtual and cloudable**
- **Assemble** some artefacts into other artefacts
- All relationships between artefacts must be **modelled explicitly** – thus the system's structure is explicit
- All models must be made to be **machine-executable** – thus the system's behaviors can be simulated in advance
- Adjust the artefacts and models to achieve the optimal behaviors for **emergent** characteristics

# Different stakeholders have different views and concerns

- Strategy
  - top manager
- Business
  - manager
  - process owner
  - super-user
  - user
- Project
  - manager
  - business analyst
- IT
  - manager
  - enterprise IT architect
  - solution architect
  - developer
  - operator



# Communication to stakeholders

- Explain to each group of stakeholders
  - Artefacts under their control
  - Relationships under their control
  - How to address their concerns (i.e. carry out a particular potential change)
- Example
  - architectural framework for digital repeatable systems
  - A comprehensive set of recommendations, models, patterns and examples of how to transform existing disparate IT systems into a coherent, agile and flexible BPM/SOA solution

# Strategy: top managers

- The architectural framework is not about how to make your products better, different and more attractive for the market place – this is for you to decide
- What it offers is to help you reduce the overheads in doing so – your flexible digital repeatable system will become an enabler for your business innovations



# Business: enterprise architects

- Help in the definition of the different types of architecture

Maturity level	Technology architecture	Data architecture	Application architecture	Business architecture		Enterprise architecture
Optimising						
Managed						
Defined						
Under development						
Initial						
None						



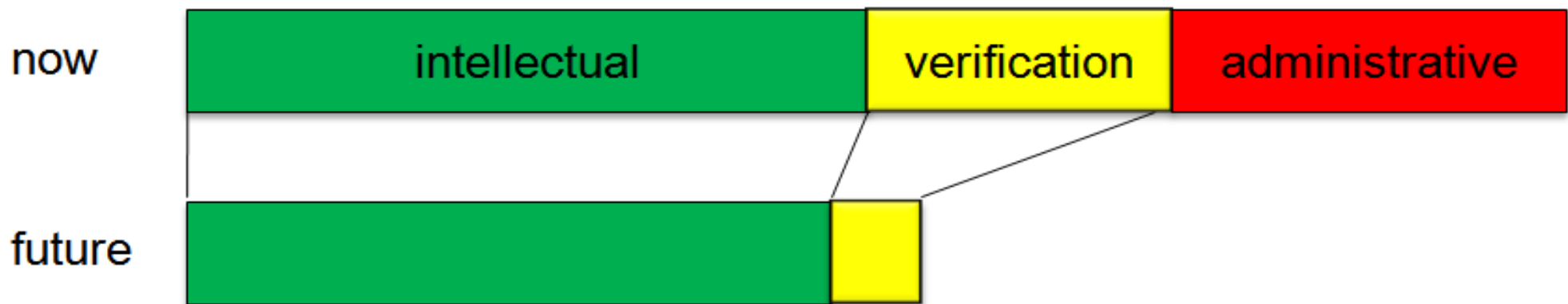
# Business: managers

- The architectural framework goal is to help you to streamline your critical business processes by
  - automating their management
  - eliminating work which does not add value
  - integrating existing applications around the business needs
  - evolving information systems in a coordinated manner
- Should make use of the synergy that exists between business needs and IT potentials



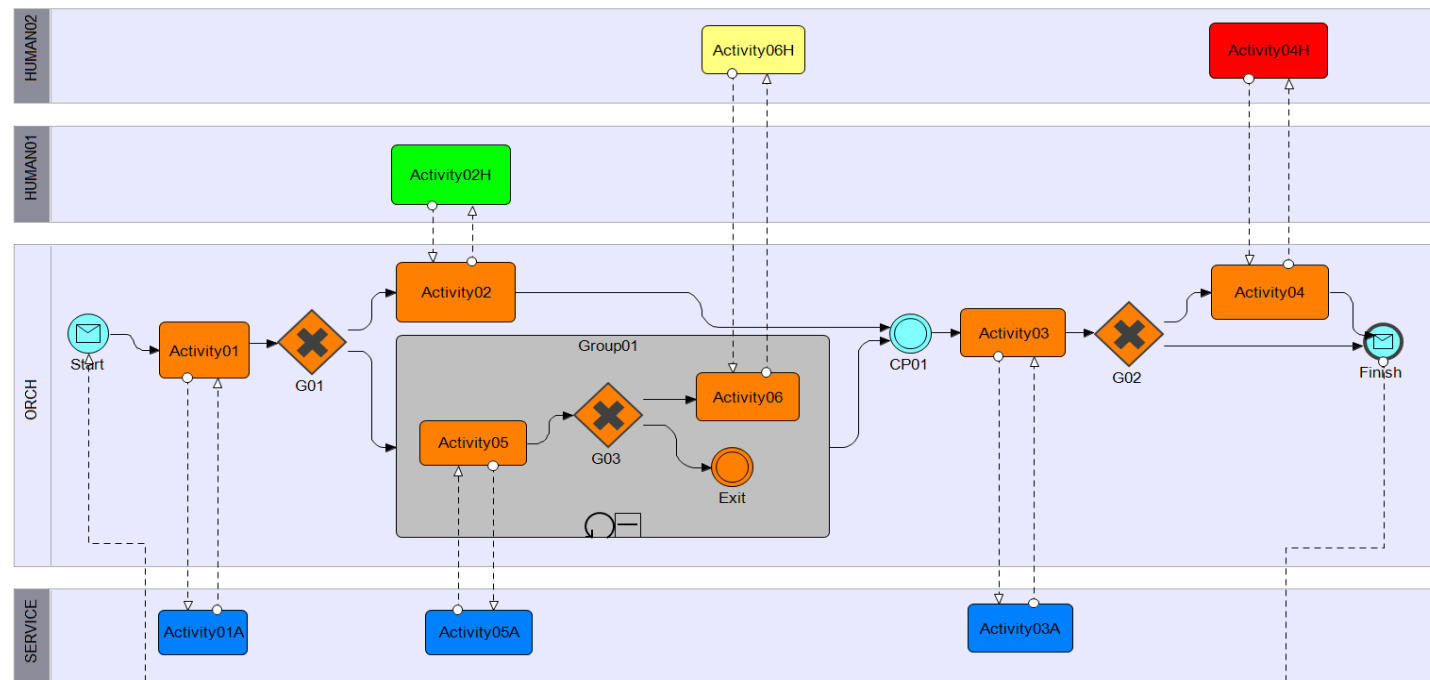
# Business: process owners

- The architectural framework classifies all human activities as intellectual (evaluation, decision-making, etc.), verification or administrative
- The goal is that the humans should perform only intellectual activities, and other activities should be automated (which may also improve quality)



# Business: super-users

- Proactive control over execution of business processes
- Delegation of complex tasks to less-qualified staff members
- Some maintenance without systematic involvement of the IT



# Business: users

- Common dashboard (over different applications) with tasklist, worklist, notifications
- Common approach for the implementation of different solutions



N°	Titre demande	Branche	Domaine	Application	Composant	Etat	Date demande	
25	Création d'un nouveau produit mixte à versements	Vie				en saisie	2004.10.23	Ouvrir
38585	PE attribuée sur des contrats désactivés	Mal. Lamal				validée	2005.06.05	Ouvrir
558	Passer le décaissement en mode interne	LPP				annulée	2006.03.02	Ouvrir
12	Période de primes en trop sur police suite diminut	Mal. Lamal				suspendue	2006.03.03	Ouvrir
5	Prime incorrecte sur police suite diminution	Mal. LCA				en développ.	2006.03.04	Ouvrir
5553	Localité 1314 Moiry (nouveau NPA : 1148)	Accident				refusée	2006.03.05	Ouvrir
55	Message d'erreur lors de la création d'une nouvelle					en saisie	2006.03.06	Ouvrir
189	Suppression de personnes saisies à double					validée	2006.03.07	Ouvrir

# Project: managers

- Achievement of common understanding within a project through clarification of the different views of artefacts
- Better visibility of artefacts
- Shorten the gap between modelling and implementation

Today

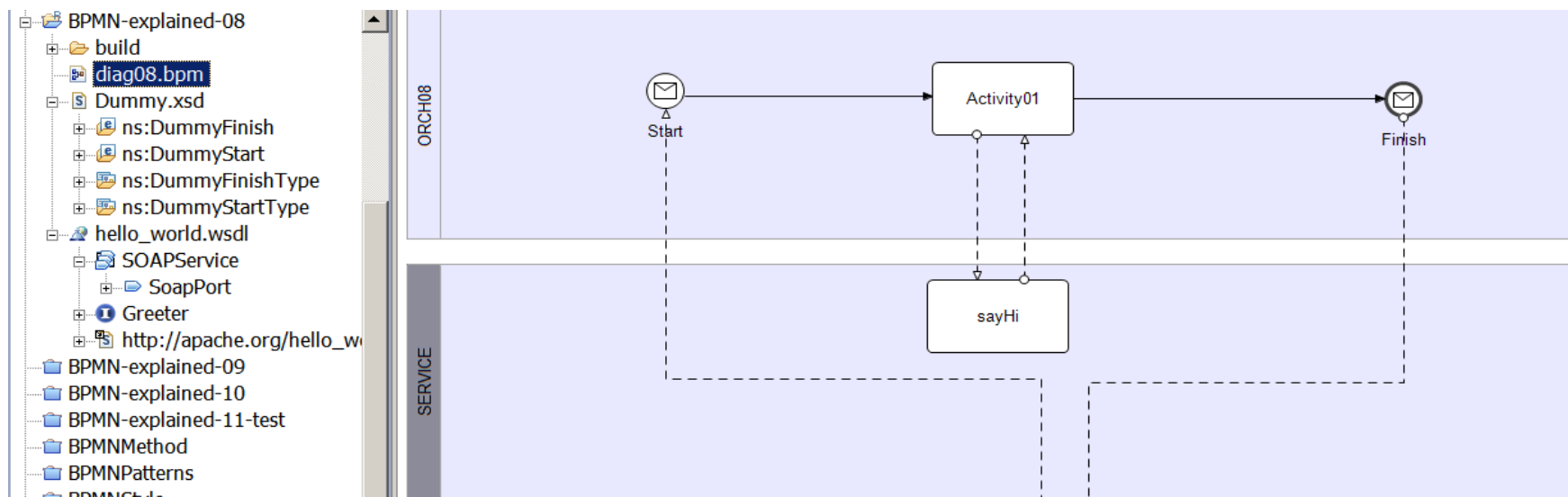


Tomorrow



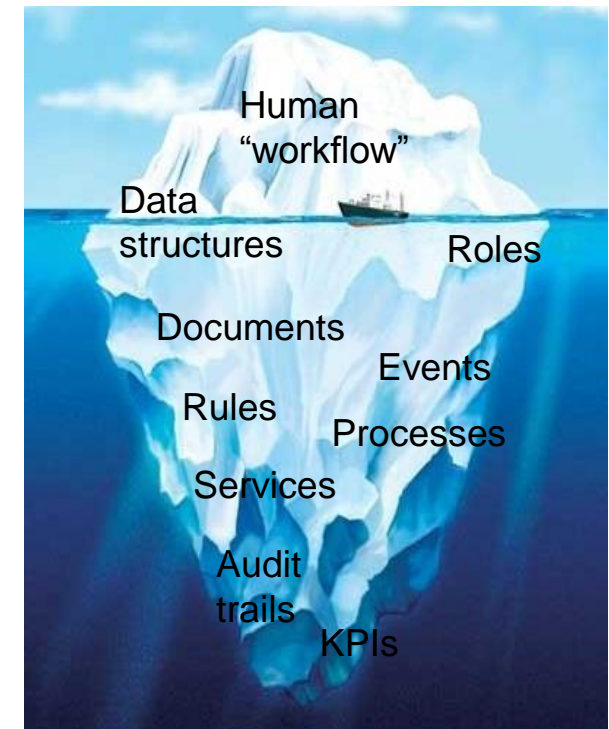
# Project: business analysts

- The architectural framework offers a modelling procedure to guide you to produce executable models
- Such a model acts as a skeleton or foundation to which the IT attaches services to obtain the implementation



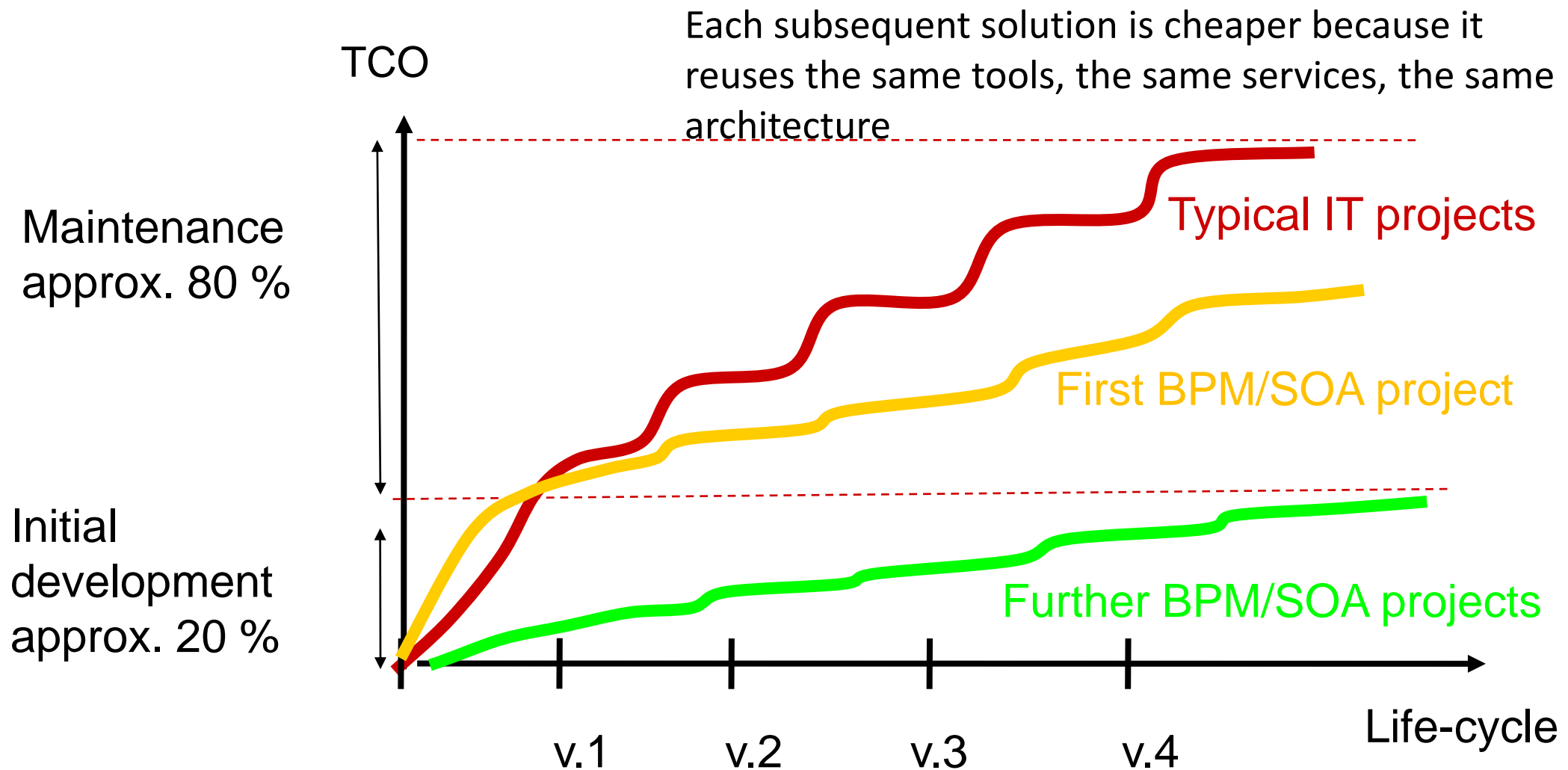
# Project: business analysts

- A modelling procedure
  - four-phase guidance to produce executable models
  - diagramming style
  - naming conventions
  - several practical patterns
- Promoting joint work between the business and IT
- Quick iterations for building an operational prototype



# IT: managers

- Considerable reduction of TCO



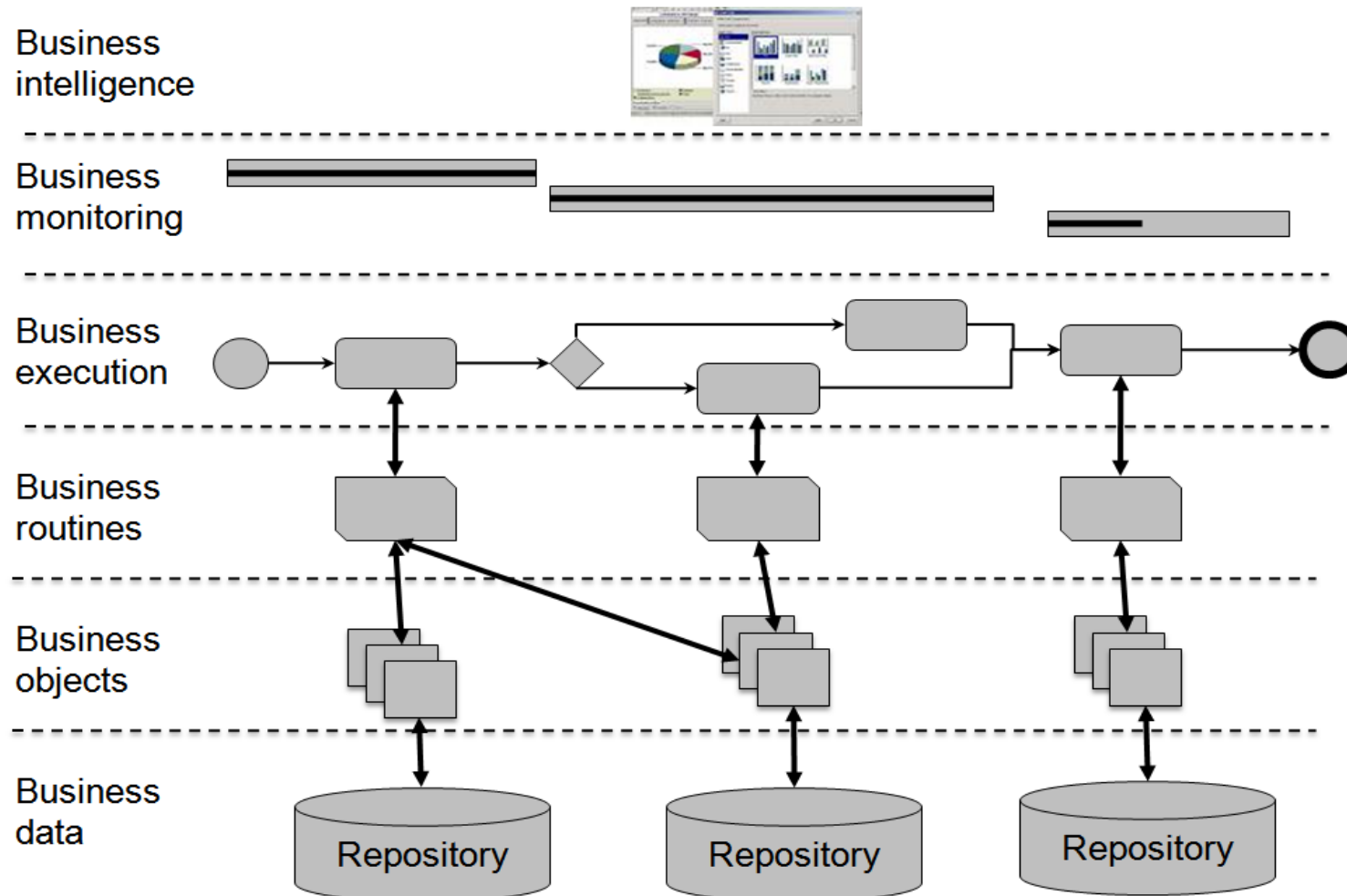


# IT: enterprise IT architects

- Architected flexibility – your digital repeatable system is easily adaptable to practically all aspects of the organisation
  - policies and priorities
  - constantly changing business processes
  - business innovations
  - computer knowledge and culture of the users
  - IT systems
  - size and complexity
  - data
  - SLA

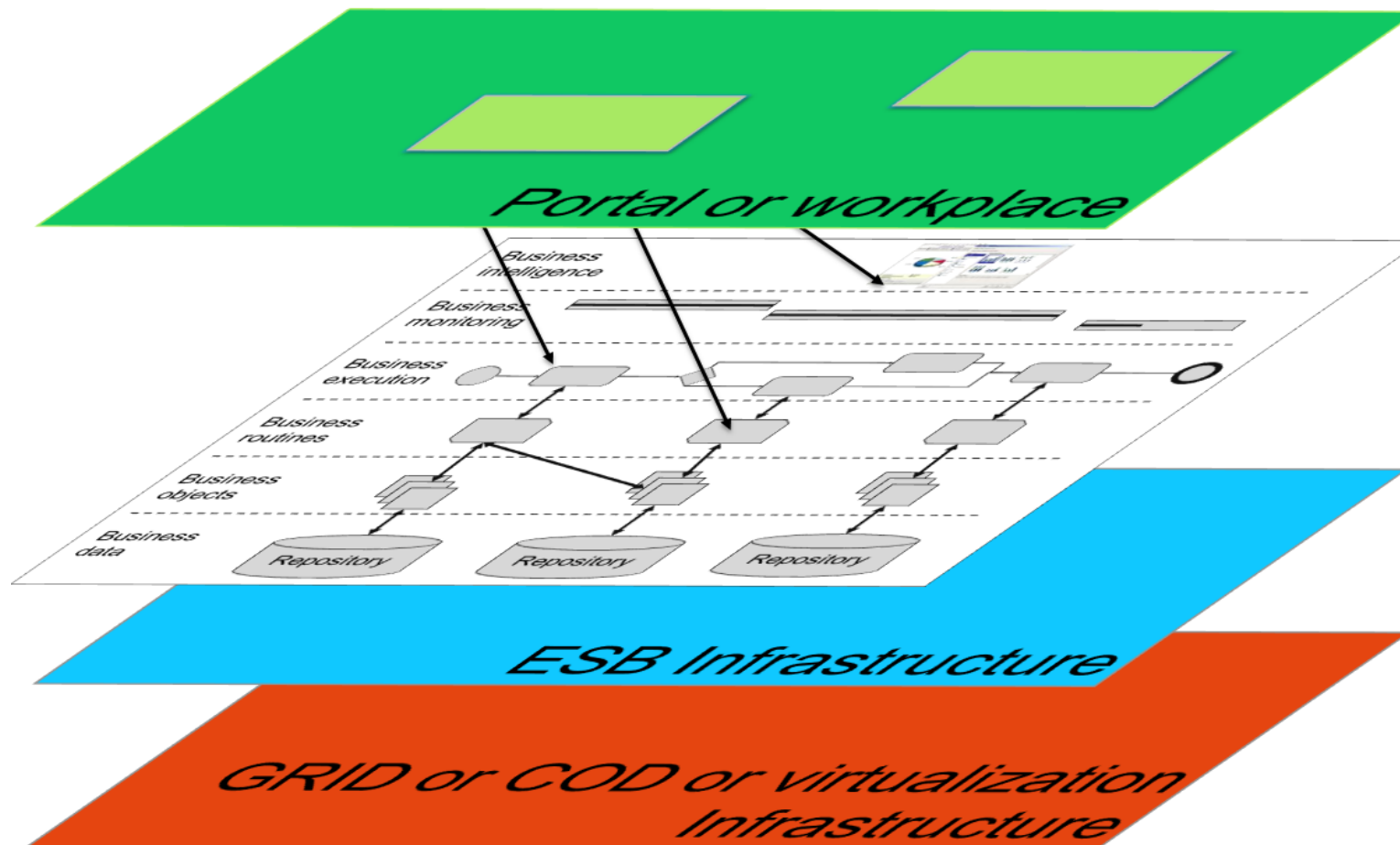
# IT: architects

- Implementation layers of artefacts



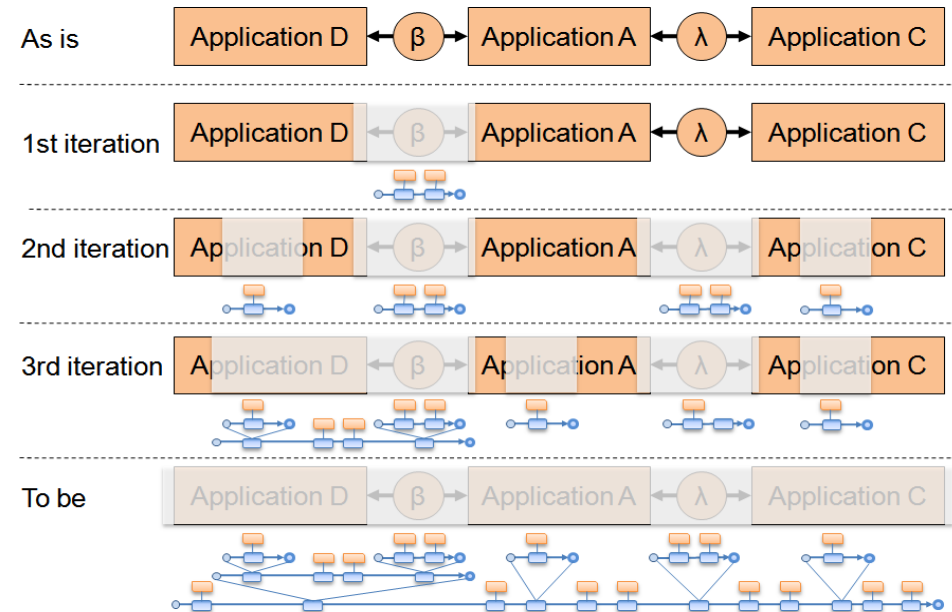
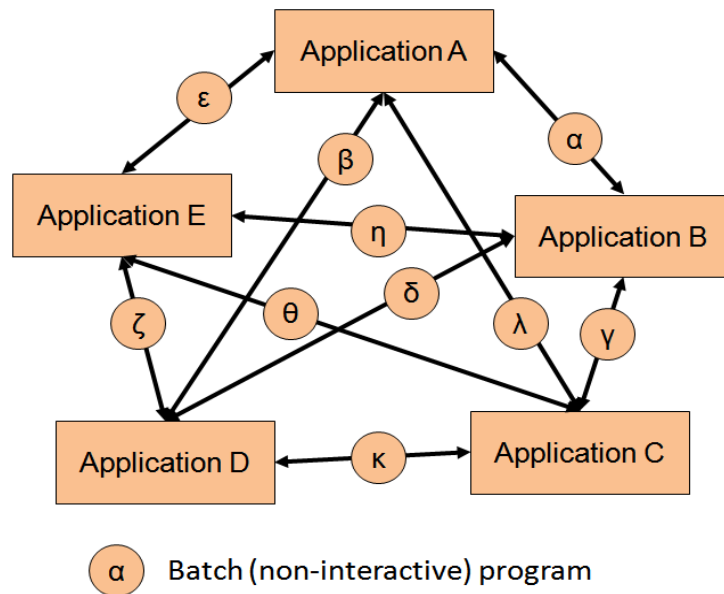
# IT: architects (cont.)

- Relationship of BPM/SOA with other technologies



# IT: developers

- Transformation from typical inter-application data flows to end-to-end coordination of services



# IT: operators

- The architectural framework helps to manage the complexity of a mixture of interconnected and interdependent services by making explicit all relationships between services
- It thus allows a correct evaluation of the availability of business-facing services from the known availability of technology-related services

# Human dynamics is critical

Variant	Strategy		Business			IT			
	Top managers	Enterprise architects	Line managers	Super-users	Users	Managers	Solutions architects	Developers	Operators
A	≈	▒	▒	▒	—	≈	▒	▒	Ω
B	≈	≈	Ω	Ω	—	↓	▒	■	©
C	—	—	↑	—	—	—	↓	—	—
D	?	—	—	—	—	—	—	—	—
E	—	—	?	—	—	—	—	—	—
F	—	—	—	—	—	?	—	—	—
G	—	?	—	—	—	—	—	—	—

Marker	Stakeholder position relative to the BPM system
↑	Process is the most important artefact (or, just create processes)
↓	Service is the most important artefact (or, just create services)
≈	Do whatever you want as long as you make my life easier in the end
Ω	Do whatever you want as long as I don't have to do anything extra
▒	I agree on the whole with the chief architect of the BPM system
?	I don't know how to improve our system
—	Stakeholder position unknown
©	My functional unit will invent something better
■	"Business as usual"; I will continue working as before

# Questions?

- E-mail: [alexandre.samarine@gmail.com](mailto:alexandre.samarine@gmail.com)
- Mobile: +41 76 573 40 61