



**Systemic Framework for
Enterprise Architecture & Transformation**

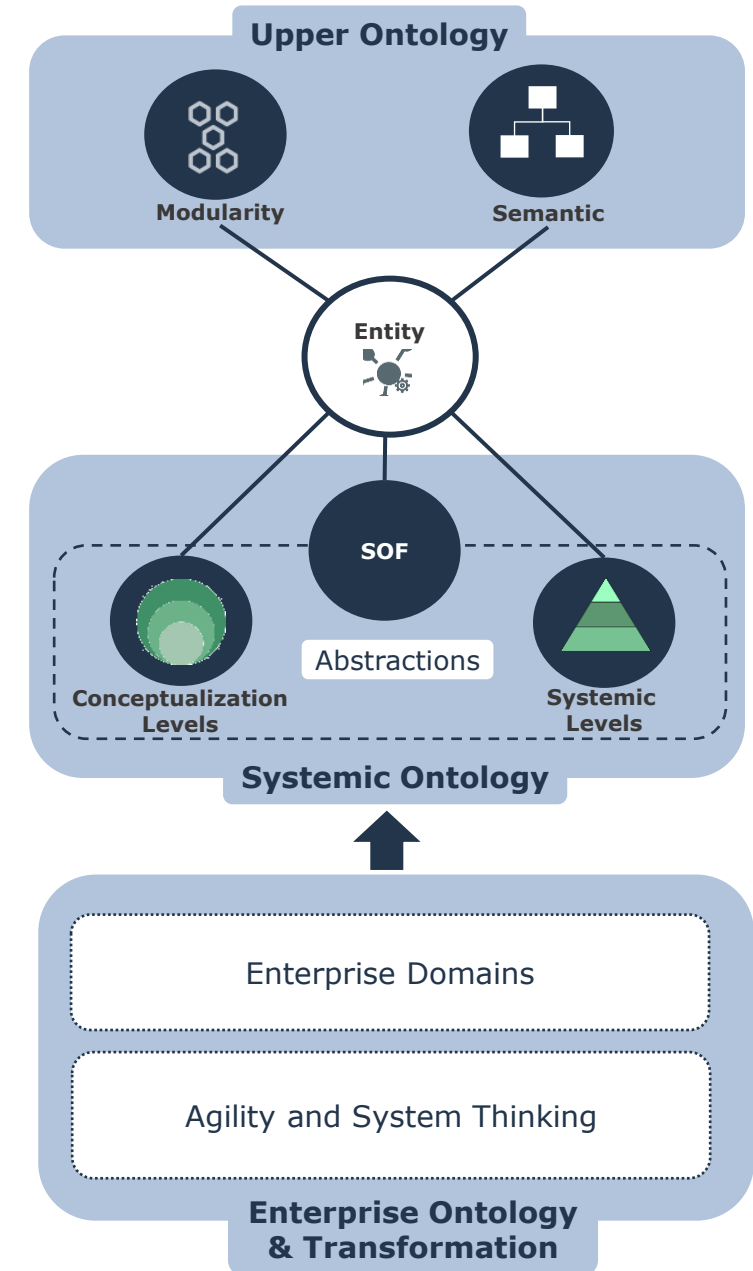
Enterprise Architecting

Introduction

- This document is an integral component of the SEAF architectural framework. It provides foundations to address the challenges posed by Enterprise Architecture in the 21st century, which include :
 - Increasing complexity in system structures and behaviors.
 - Growing intricacy in architecture, management and governance of these systems.
 - The mission of the framework is to demystify these complexities, ensuring they are comprehensible to a broad audience, thereby facilitating the design and management of complex-systems across all scales, from micro-systems to enterprise level systems.
- Enterprise Architecting refers to the overarching method and conceptual framework used to describe, understand, and communicate the complex structures and dynamics of an enterprise.
- It integrates both the operating aspects of the enterprise (how it functions and interacts within its ecosystem), the transformational aspects (how it evolves and sustains over time through initiatives, asset management) and how these transformations are governed to ensure effectiveness, efficiency and reliability.
- The following slides present the foundations of enterprise architecting methodology.

Foundations of enterprise modeling

- **Modularity** provides the syntax for building robust, manageable, and scalable architectures, based on the principles of [compositionality](#) and [packaging](#).
- **Semantic** provides robust capabilities for classifying and composing entities, from time-bound entities ([individuals](#)) to [families of concepts](#), enabling effective representation of meaning.
- The **Systemic Operating Framework (SOF)** serves as the overarching language that describes why and how a system [operates and interacts](#) within its ecosystems.
- **Abstractions** organizes systems and concepts in degree of abstractions, including [systemic levels](#) and [conceptualization levels](#).
- **Enterprise Domains** formalize the various disciplines that make-up EA, ranging from [enterprise road-mapping](#) to [System ArcDevOps](#).
- **Agility and System Thinking** ensure that the enterprise evolves and sustains over time through governed initiatives, architected for flexibility and responsiveness in complex and dynamic business environments.



Enterprise Architecting overview

What is it?

What is it used for?

How to build it?

Enterprise as a special case of system

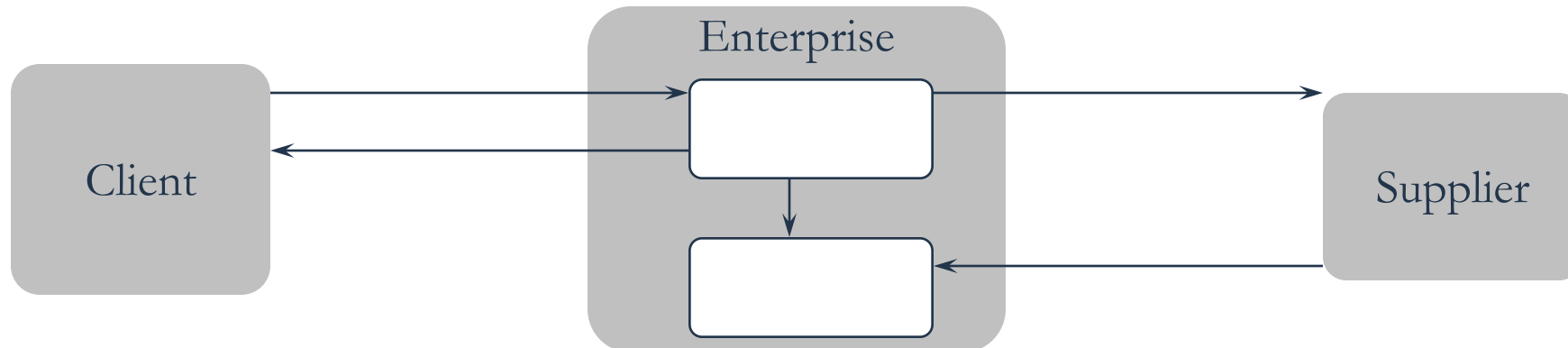
- We model the Enterprise
- ... but what do we mean by Enterprise ?
 - Aren't ERPs modeling the entire scope enterprises' activities?
 - Cannot we find all members of the enterprise in HR Systems?
 - What is missing in services provided by Asset Management tools?
 - ...
- Our goal is to model **how the Enterprise "Functions/Operates"**, identifying its objectives, the resources (Assets) it uses and organizations it sets up to achieve them, as well as putting in place the governance that ensures that everything is progressing as intended.
- To accomplish this, we have adopted an approach grounded in Systems Thinking.

Modeling the enterprise – System thinking

- Systems thinking is an interdisciplinary approach that seeks to understand systems as organized and complex wholes. It focuses on the relationships and interactions between the elements of a system rather than on the elements themselves.

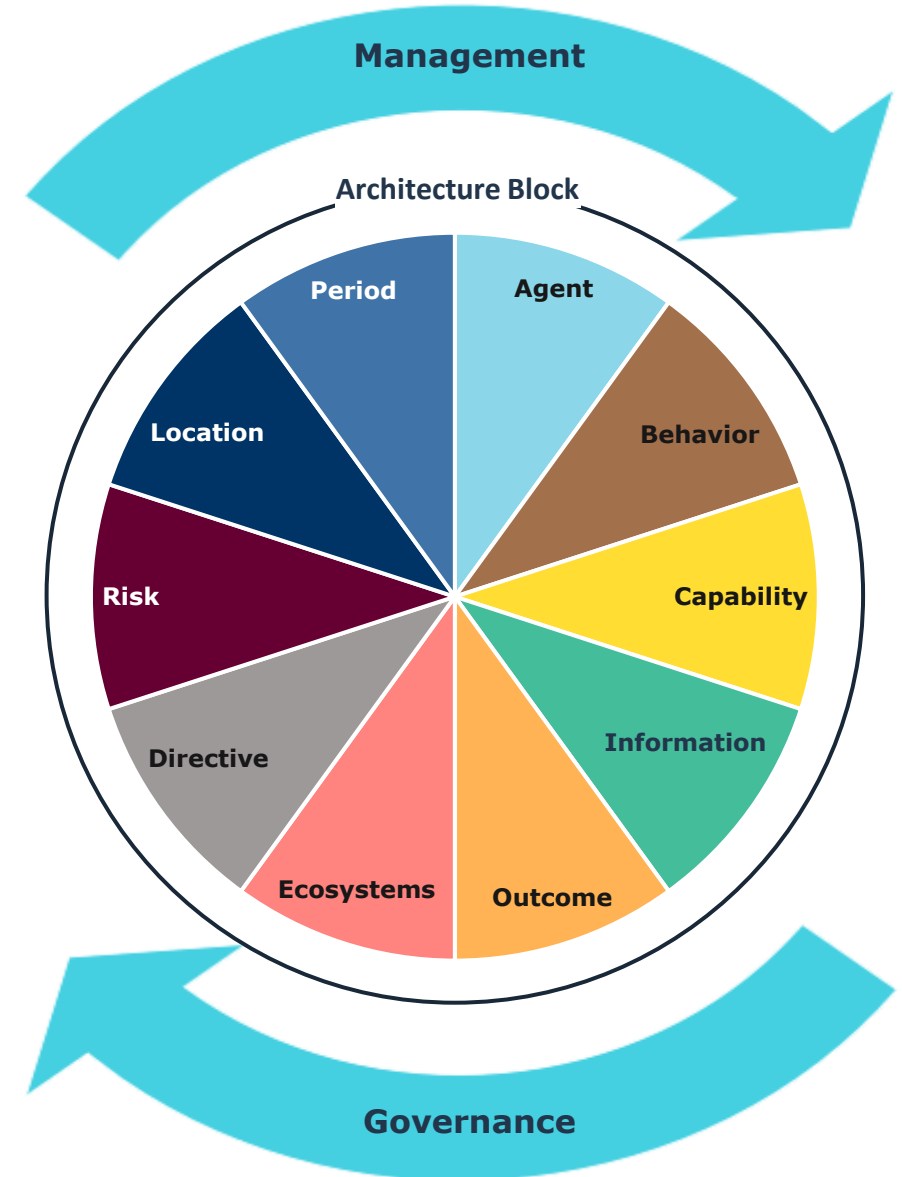


- It enables a black-box/white-box approach of the various aspects of the functioning enterprise.
- These aspects are explored in the following slides.



Summary of aspects of a functioning enterprise

- The various aspects that explain why and how an enterprise “functions” are outlined in the diagram on the right.
- Further details are provided in the following slides.

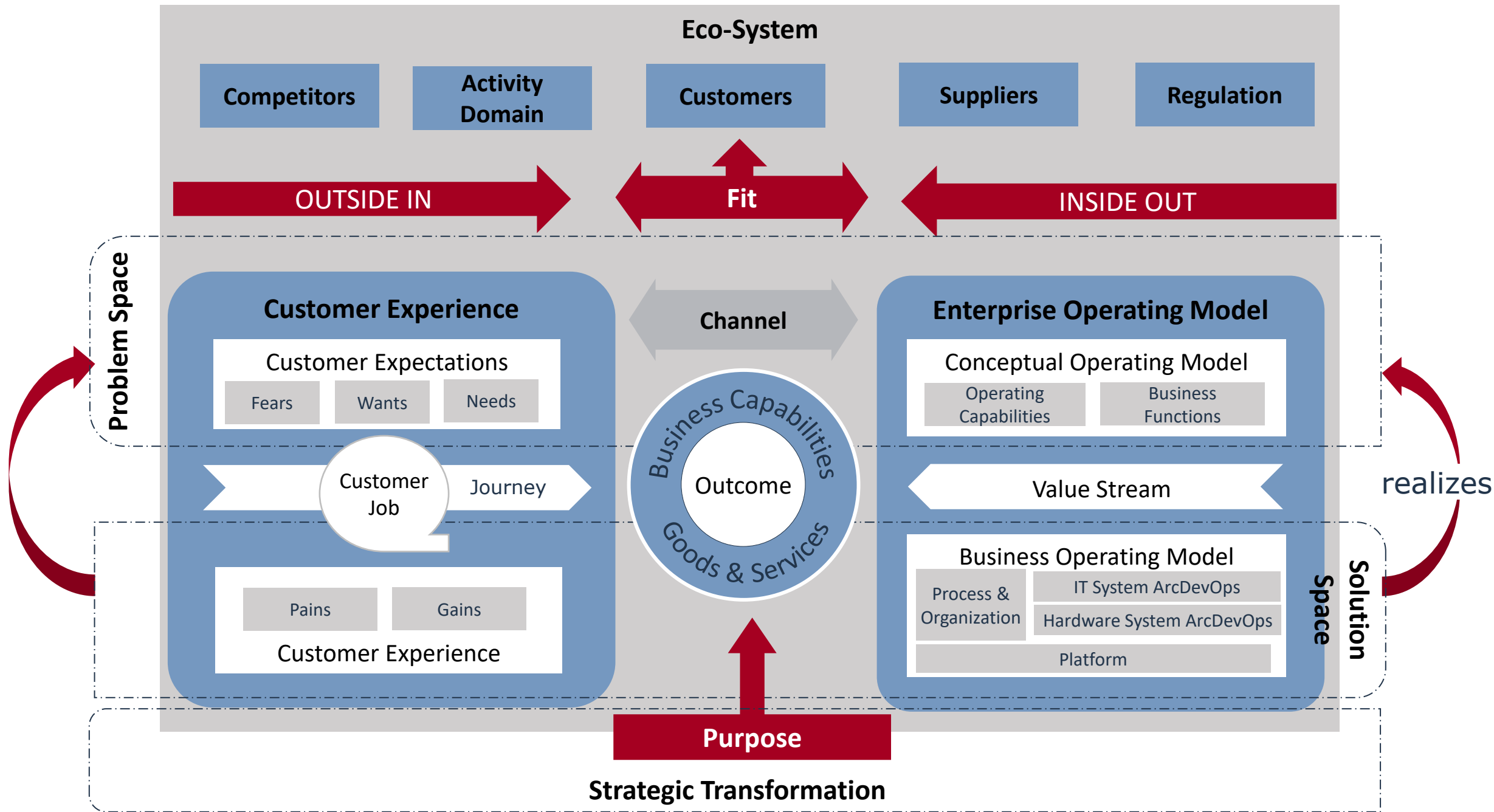


Types of system and business functions

- Types of systems
 - Deterministic: systems in which neither the parts nor the whole are purposeful.
 - Animated systems in which the whole is purposeful, but the parts are not.
 - Social: systems in which both the parts and the whole are purposeful.
 - Ecological systems in which some of the parts have purposes but not the whole.
- Activity Domain
 - See "Social" in Types of systems. It has to be "Social"
- Business function
 - See "Deterministic"
 - See "Animated"
 - See "Social"
 - See "Ecological"

Capability Models & Operating Models

- A key aspect of the method is the distinction between:
 - User journeys which express customer outcome expectations in their job context.
 - Capabilities which express abilities to produce expected outcomes.
 - Operating Models which express the way agents operate to deliver expected outcomes.
- All are positioned in the Enterprise Eco-System where:
 - Capabilities are offered in regard to customer jobs (outside-in).
 - Operating solutions are experienced by customer doing their jobs (inside-out).
 - Enterprise drivers and goals express a volition to respond to ecosystem expectations and experience, overtime.



Conceptual Operating Model

What is it?

What is it used for?

How to build it?

Start and stop criteria (Activity Domain)

- Why do we use a Business function ?
 - Because we to model something that is composed of a sub controlling systems over another sub execution system
- Start (why do we decompose an Activity Domain ?)
 - Finality : Can the business function select the capabilities it delivers ? (the capabilities can vary through time)
 - Is there a sub controlling systems over sub execution system ?
- Stop (why do we stop to decompose an Activity Domain ?)
 - criteria : Does the subparts have an internal management function ? If **not** then it is a **business function**, it **yes**, it is an Activity Domain

Start and stop criteria (**Business function**)

- Start (why do we decompose a business function ?)
 - OKR
 - Objectif
 - Key results
 - Resilience
- Stop (why do we stop to decompose a business function ?)
 - Are we interested to model the sub business function ?
 - Are we interested in modeling the fact that business function has a sub control function to ensure quality of its outcome ? If yes it has to be an area, if no it is a business function

MAA : False, cannot
happen,

Measurement criteria (Draft)

- Hierarchy score measurement (compared to other enterprise)
- Control score measurement (is there an area without control, is there enough autonomy)

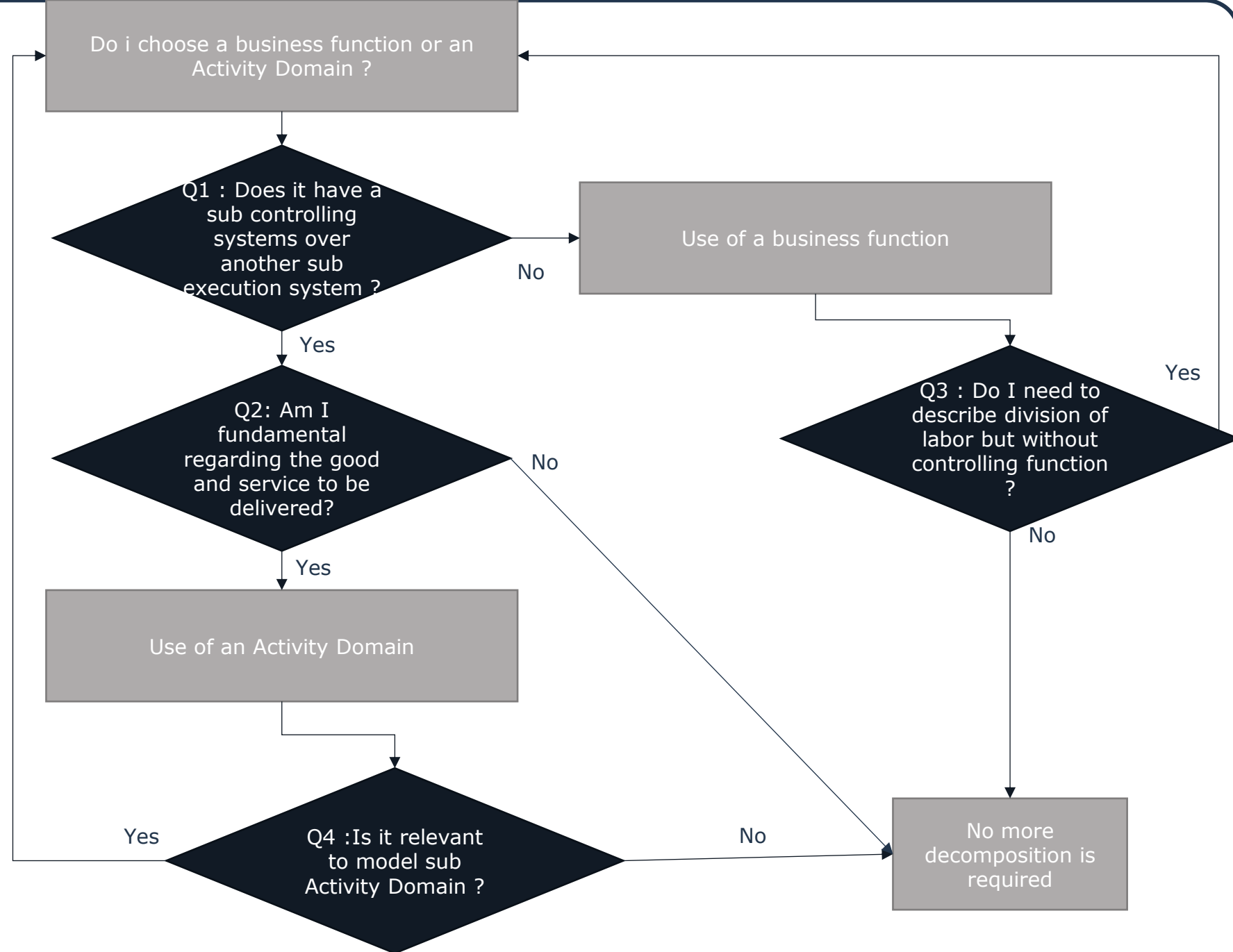
Method

Prerequisite:

- The **starting point** is always an Activity Domain
- A business function **cannot be composed** of Activity Domain
- **The purpose**, design the optimal/ideal functional division of labor that match the best way to deliver goods and services. It has to be solution independent.

Why do i need a sub controlling function ?

- Dunbar number
- Need for coordination because of Funtionnal division



Exemple 1

BFA

BF

- Subject : Defining the ideal working way of a factory, more precisely the Tests part of it. The main « agents » are production and tests and R&D. The starting agents is the **Enterprise**

- High level management :

- Q1 = False
- Q3 = False

- R&D and Production

- Q1 = True
- Q2 = False

- Tests

- Q1 = True
- Q2 = True
- Q4 = True

- Tests Control:

- Q1 = False
- Q3 = False

- Tests Execution agent

- Q1 = False
- Q3 = True

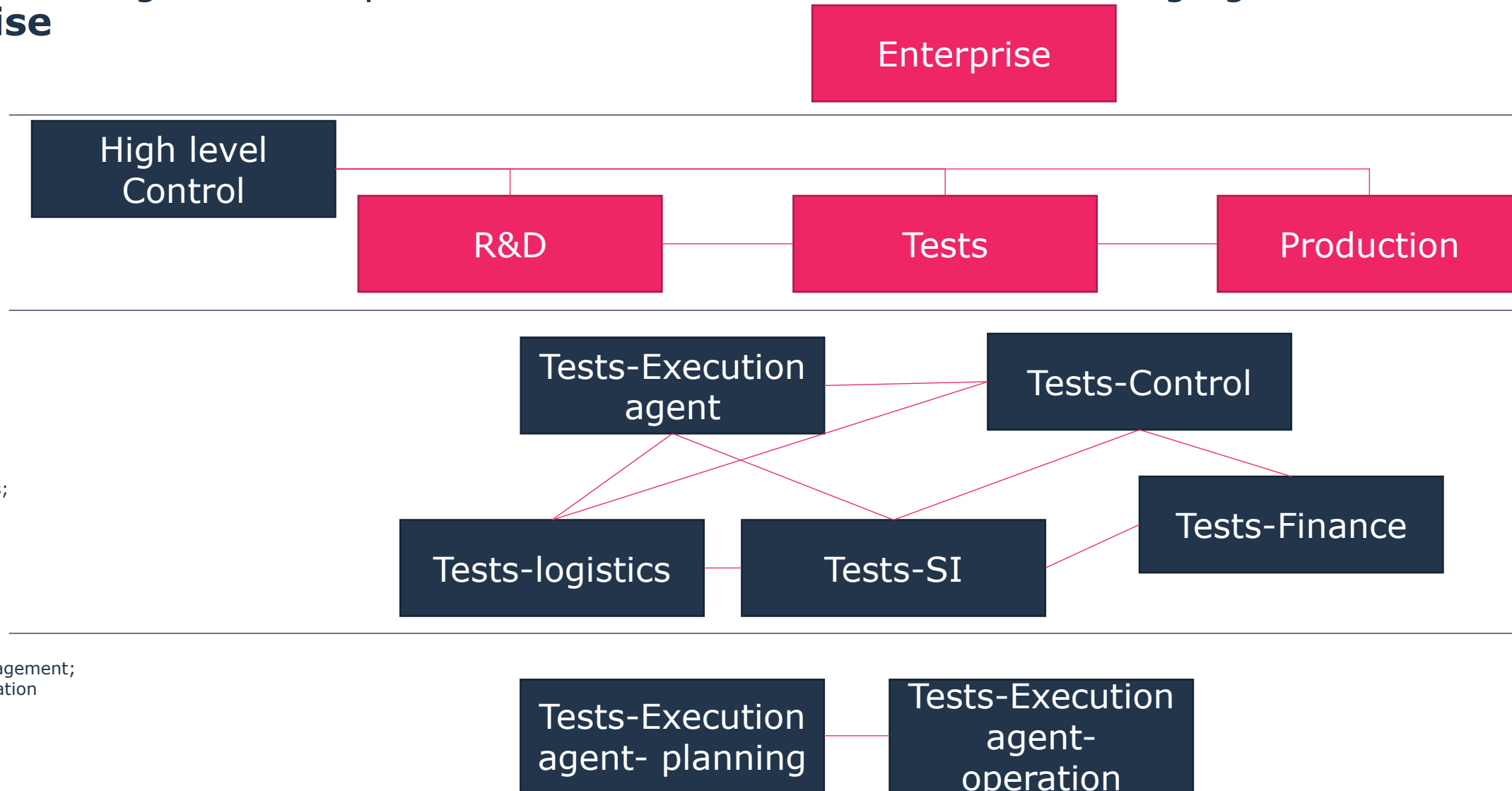
- Tests Finance; Tests-logistics; tests-DSI

- Q1 = False
- Q3 = False

- Tests Execution agent management;

- Tests Execution agent operation

- Q1 = False
- Q3 = False



Exemple 2

BFA

BF

- Subject : Defining the ideal working way of a factory, more precisely the Tests part of it. The main « agents » are production and tests and R&D. The starting agents is the **Enterprise**

- High level management :

- Q1 = False
- Q3 = False

- R&D and Production

- Q1 = True
- Q2 = False

- Tests

- Q1 = True
- Q2 = True
- Q4 = True

- Tests Control:

- Q1 = False
- Q3 = False

- Tests Execution agent

- Q1 = False
- Q3 = True

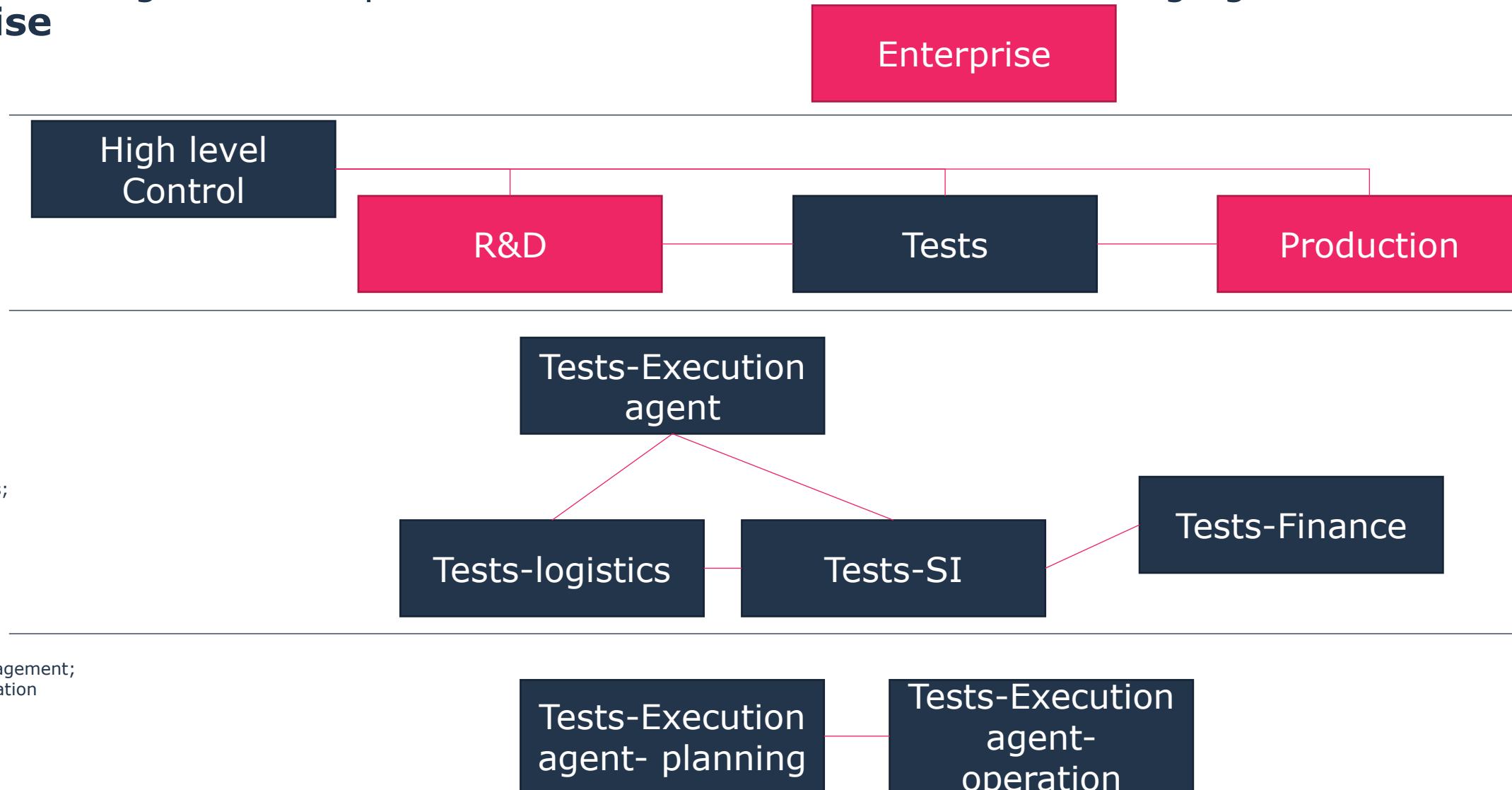
- Tests Finance; Tests-logistics; tests-DSI

- Q1 = False
- Q3 = False

- Tests Execution agent management;

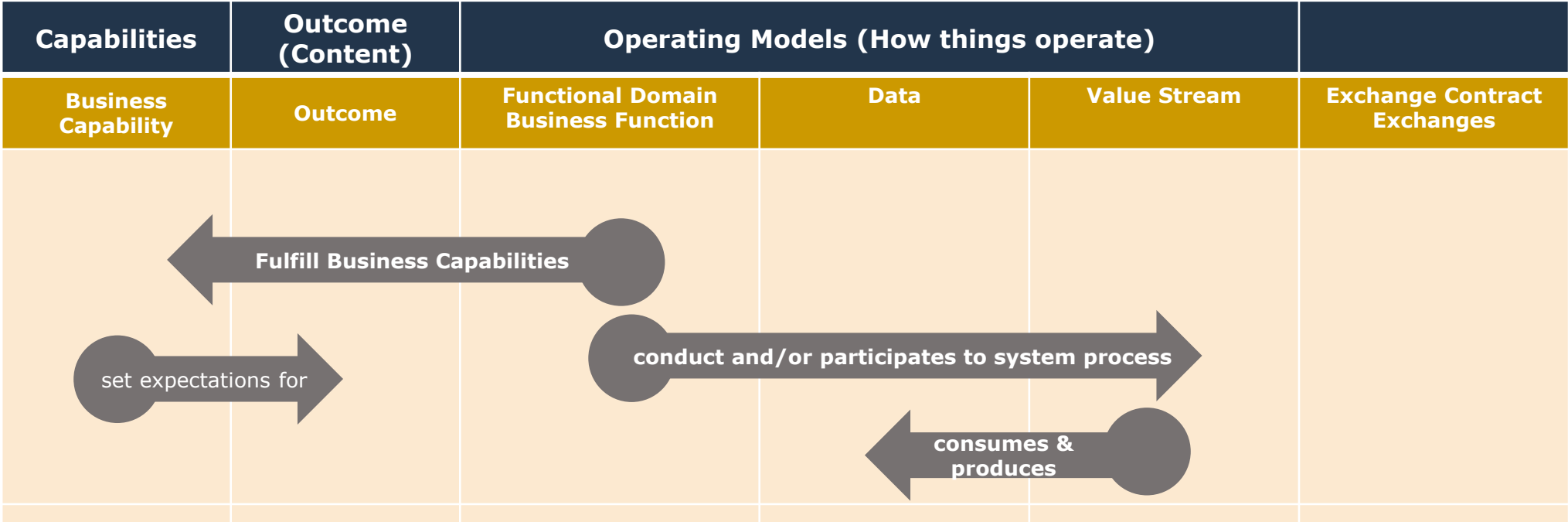
- Tests Execution agent operation

- Q1 = False
- Q3 = False



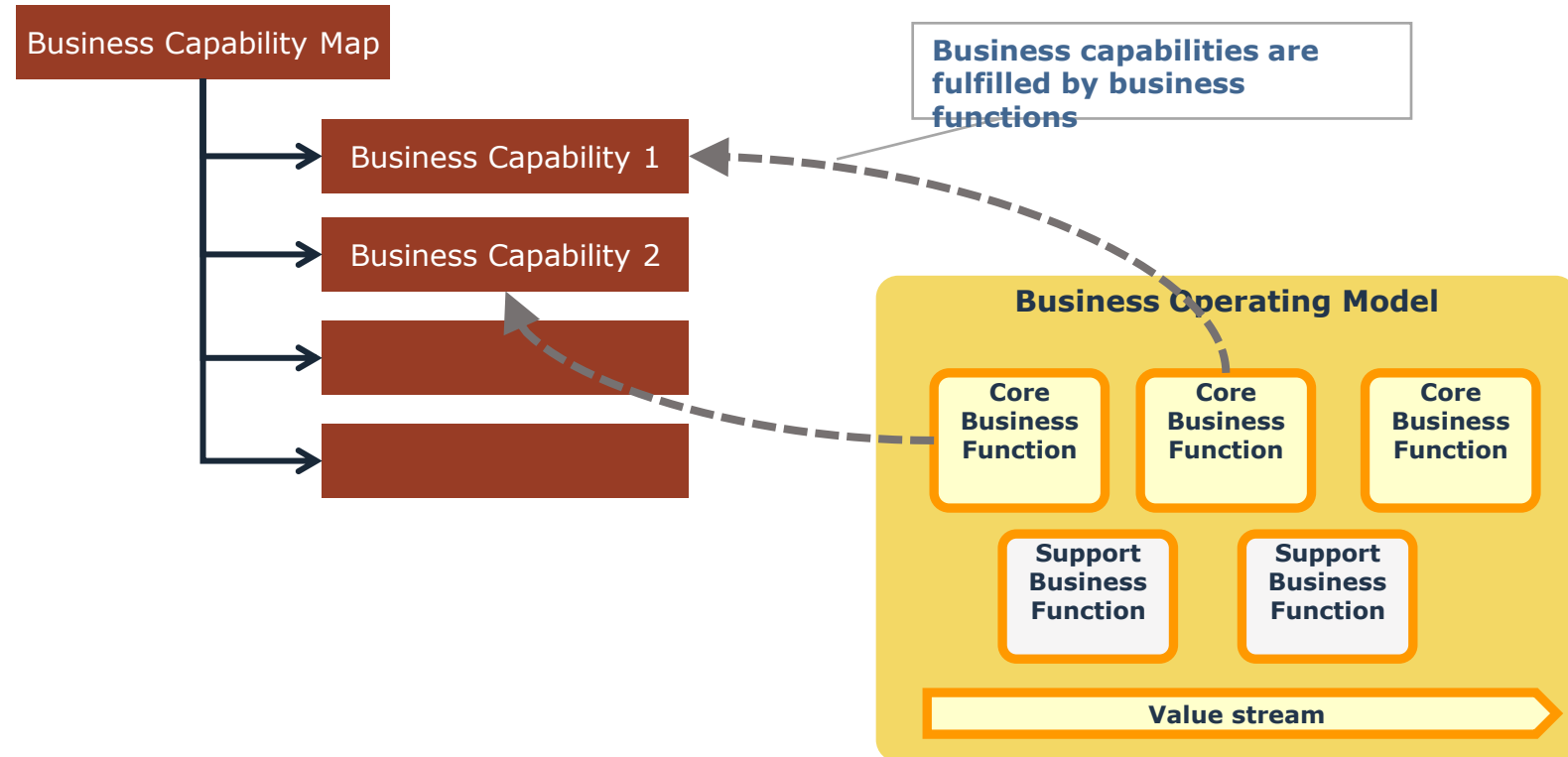
System Operating Model

- Capabilities: functionalities ability to produce and outcome.
 - Capabilities express what can be offered by Agents, without committing on how Agents behave.
- Outcomes: results, expected by Capabilities and delivered by Agents.
- Agents: who is doing, who is in charge.
- Interactions: service interactions between Agents.
- Actions: activities done by Agents (processes).



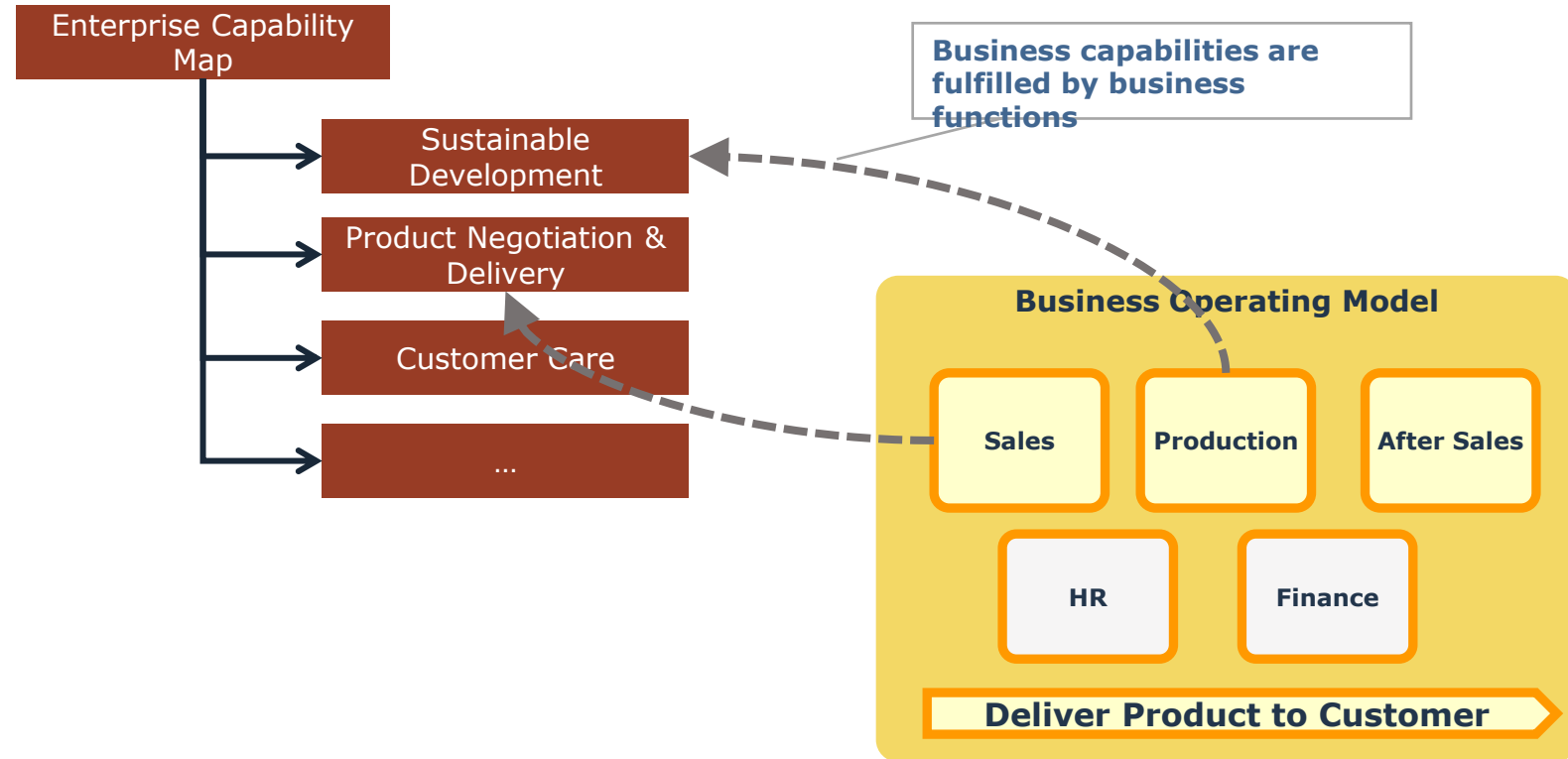
Strategic Alignment of Business Operating Model

- Business capabilities are fulfilled by business functions.



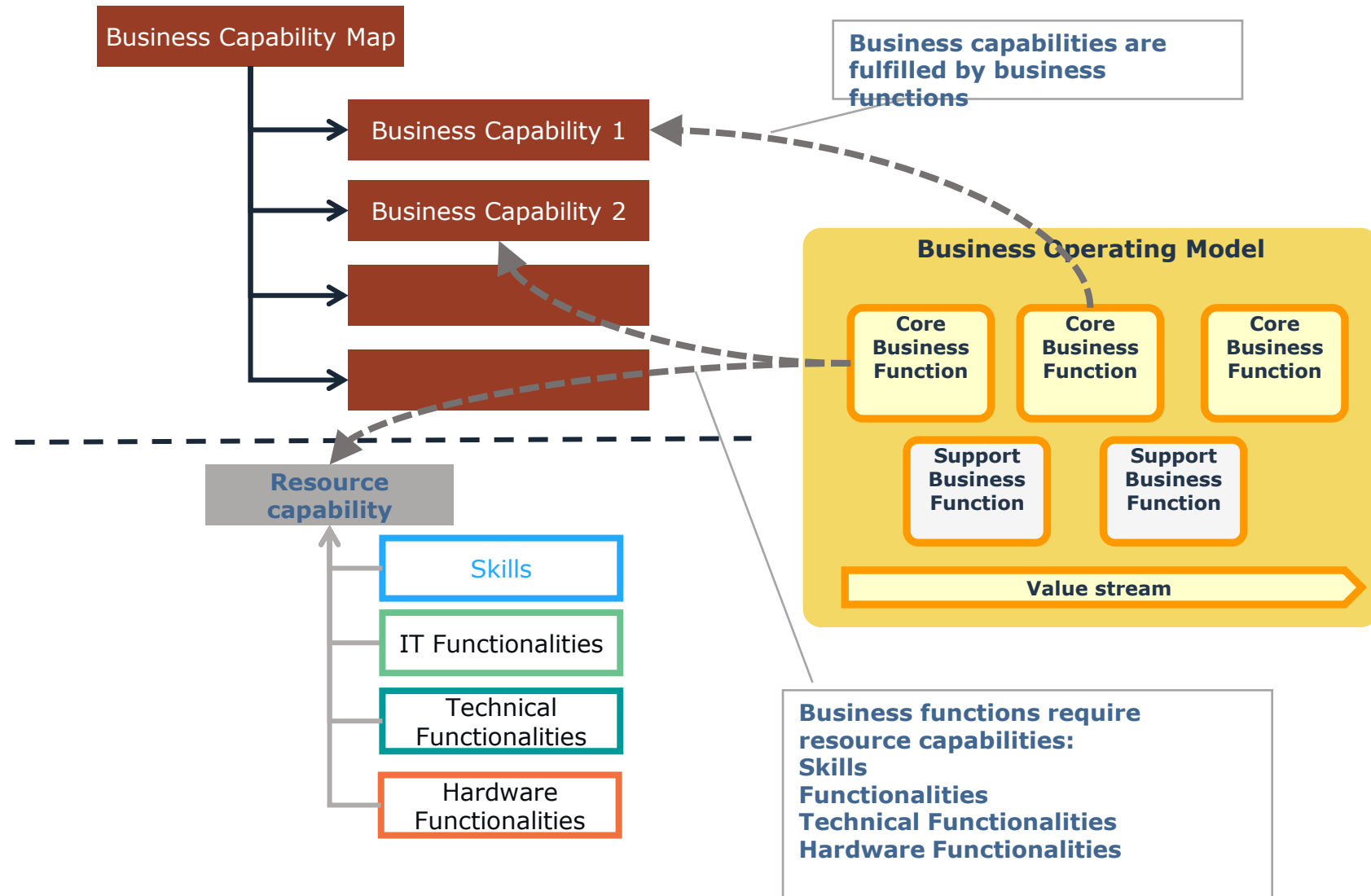
Strategic Alignment of Business Operating Model

- Business capabilities are fulfilled by business functions.



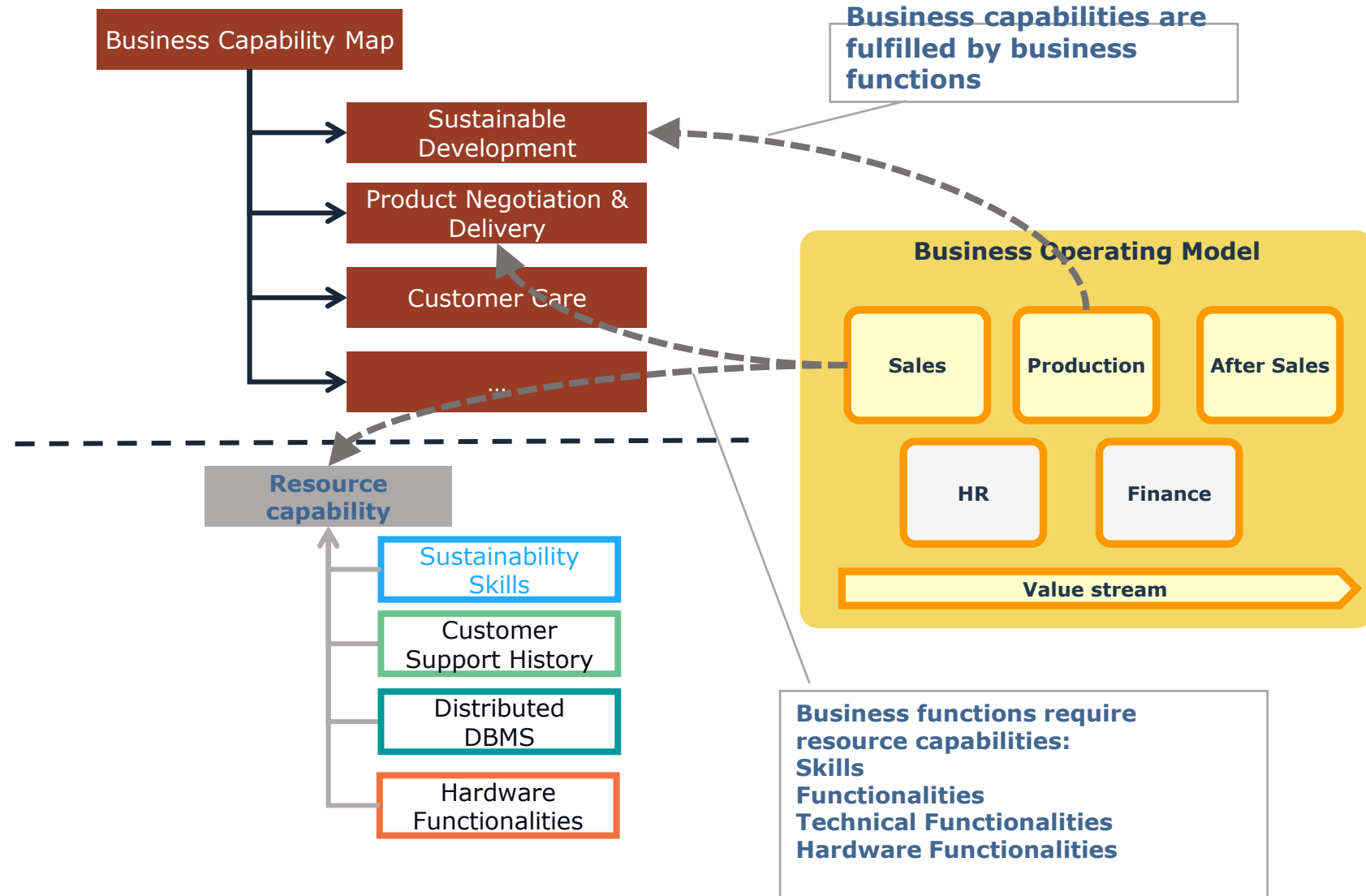
Strategic Alignment of Resources Capabilities

- Business capabilities are fulfilled by business functions.
- Business Functions required Functionalities/Skills



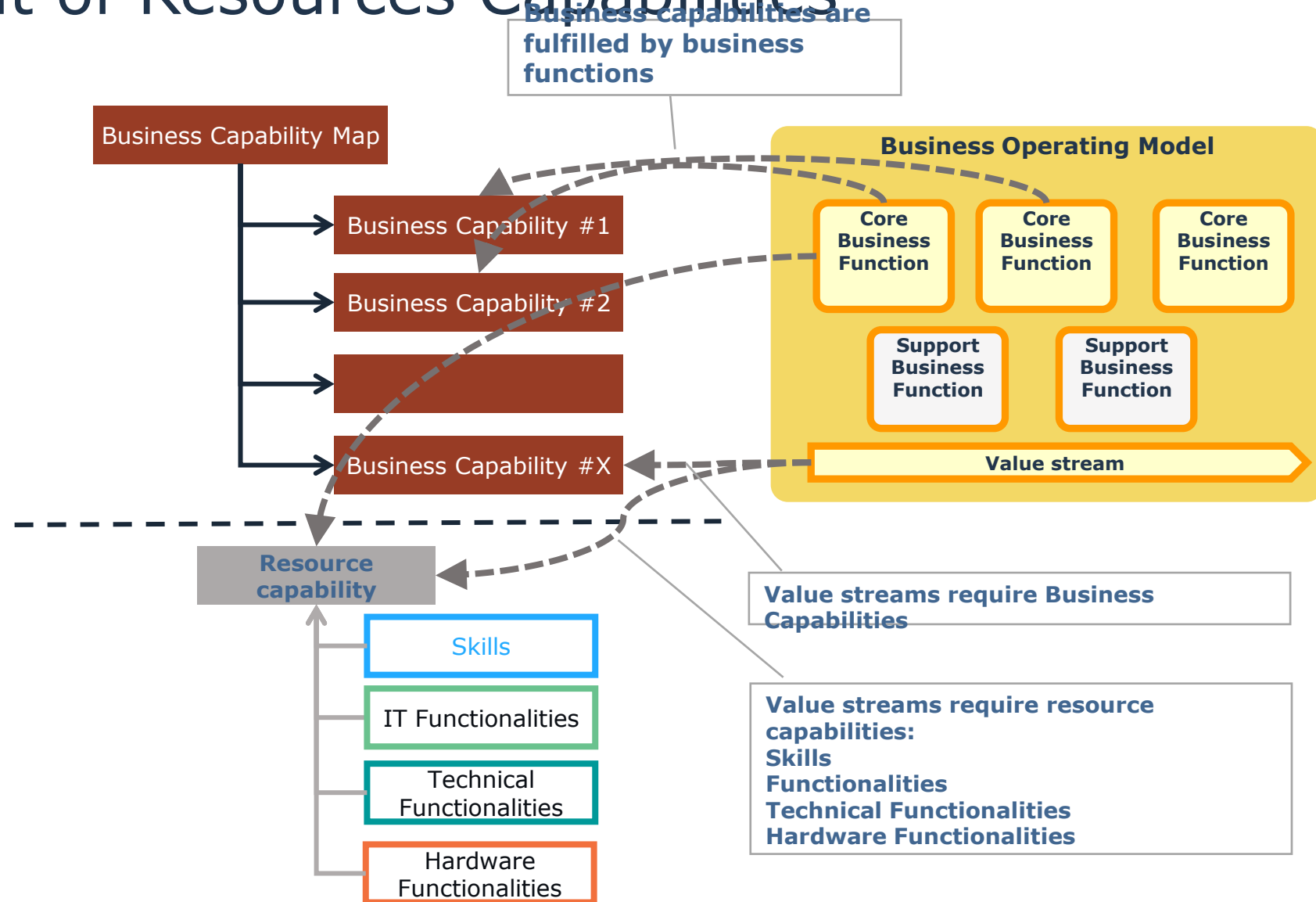
Strategic Alignment of Resources Capabilities

- Business capabilities are fulfilled by business functions.
- Business Functions required Functionalities/Skills



Strategic Alignment of Resources Capabilities

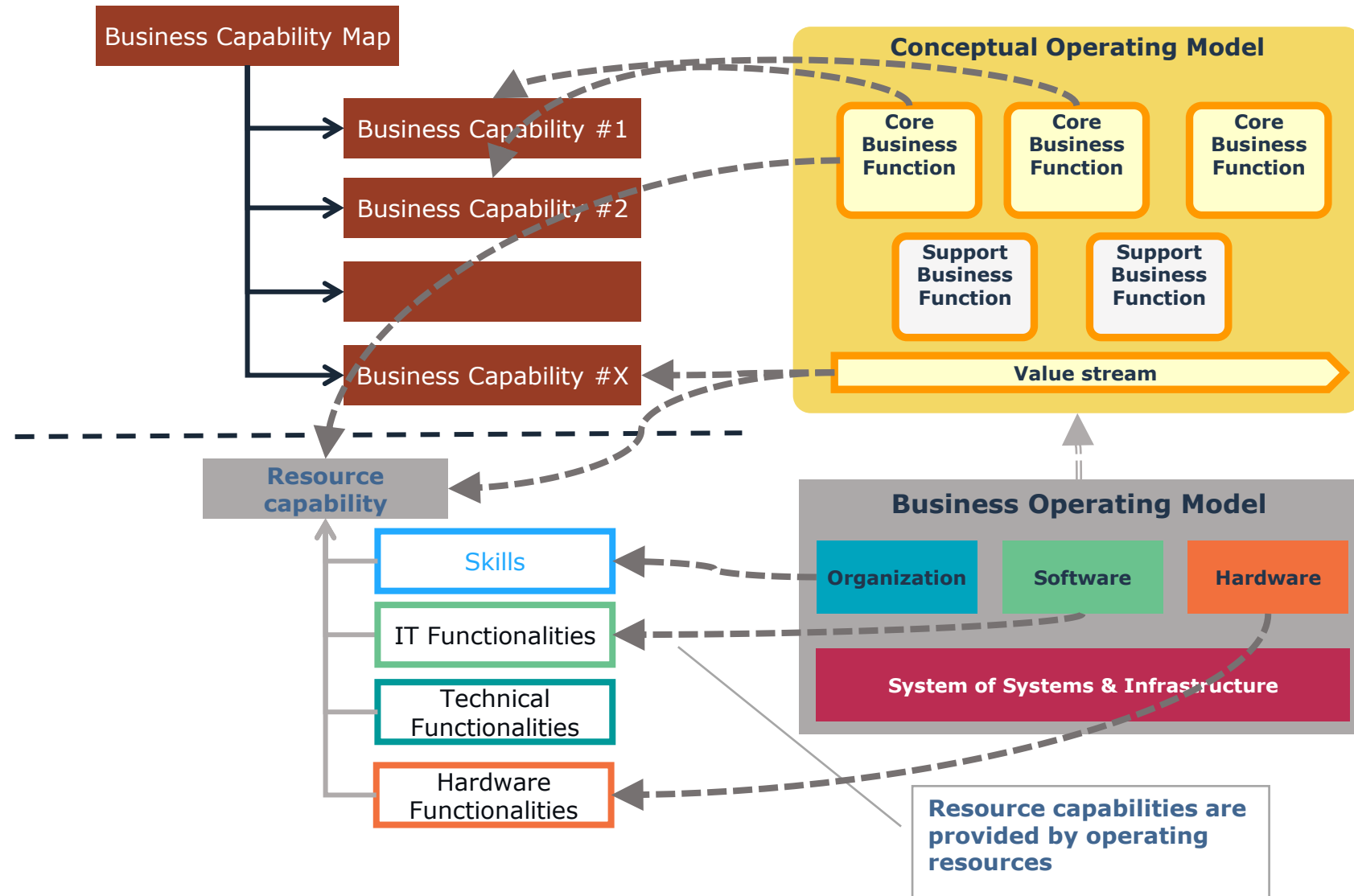
- Business capabilities are fulfilled by business functions.
- Business Functions required Functionalities/Skills
- Value Streams require Business Capabilities



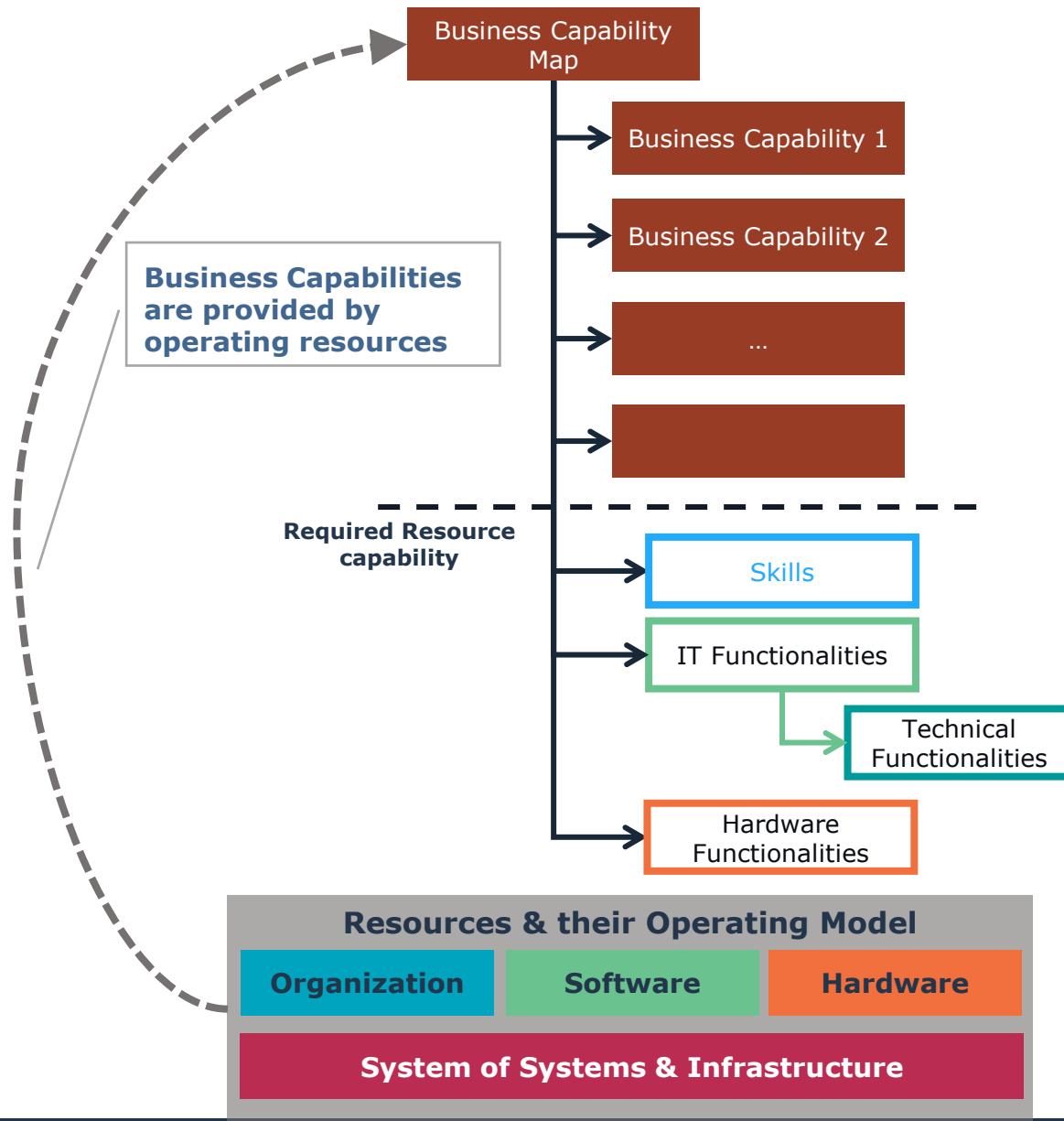
Strategic Alignment of Resources Capabilities

- Business capabilities are fulfilled by business functions.
- Business Functions required Functionalities/Skills
- Value Streams require Business Capabilities
- Business Functions and related functionalities are realized by operating Resources

→ Flexible,
modular,



Alternative Business Capability single breakdown



Capability decomposition goes down to functionalities and skills → No differentiation between enterprise strategic capabilities (business capability) and enterprise operating models (business functions).

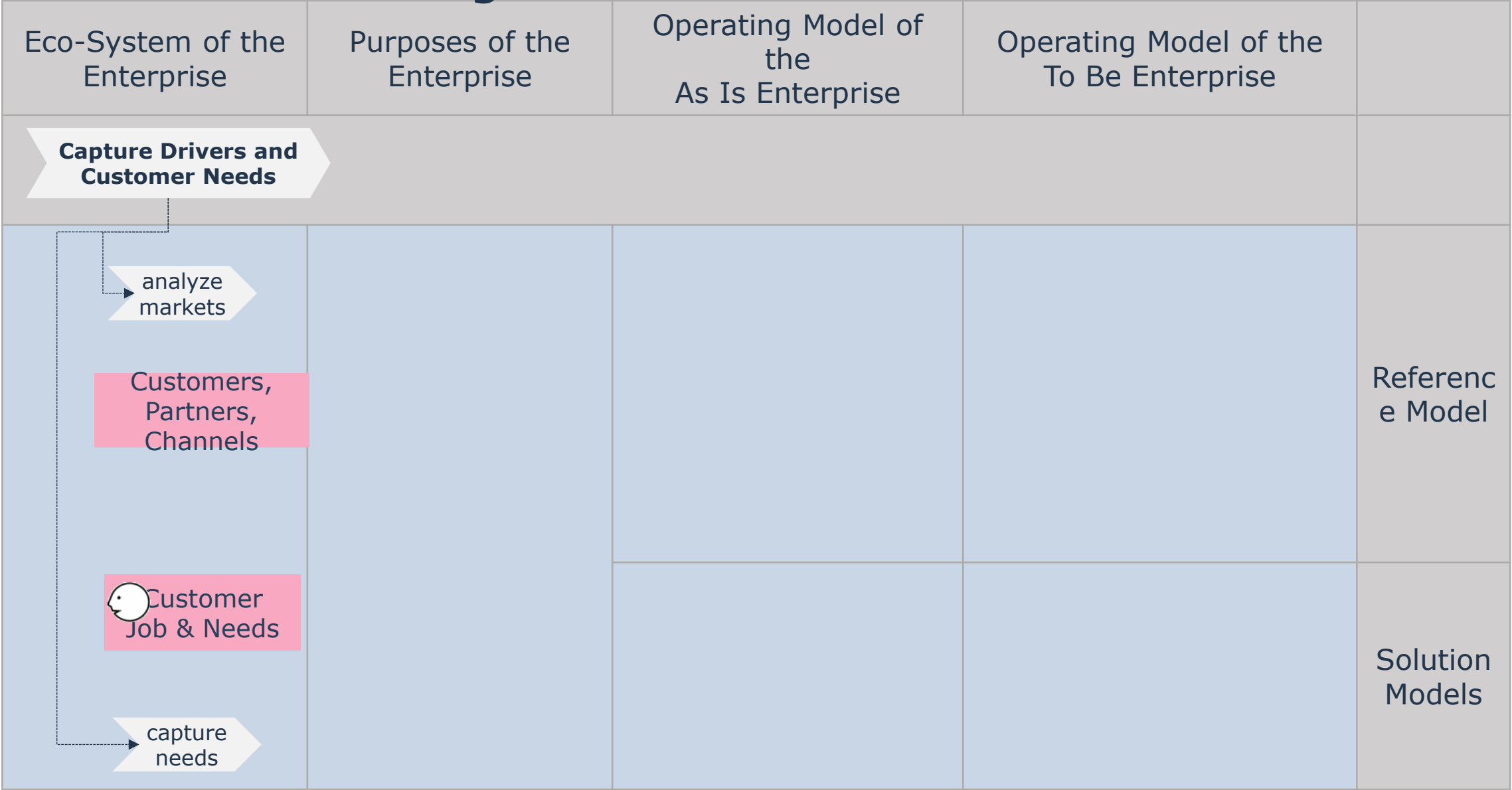
-> Preferential approach for US.

-> Major limitation: amalgamation of business capability and resource capability.

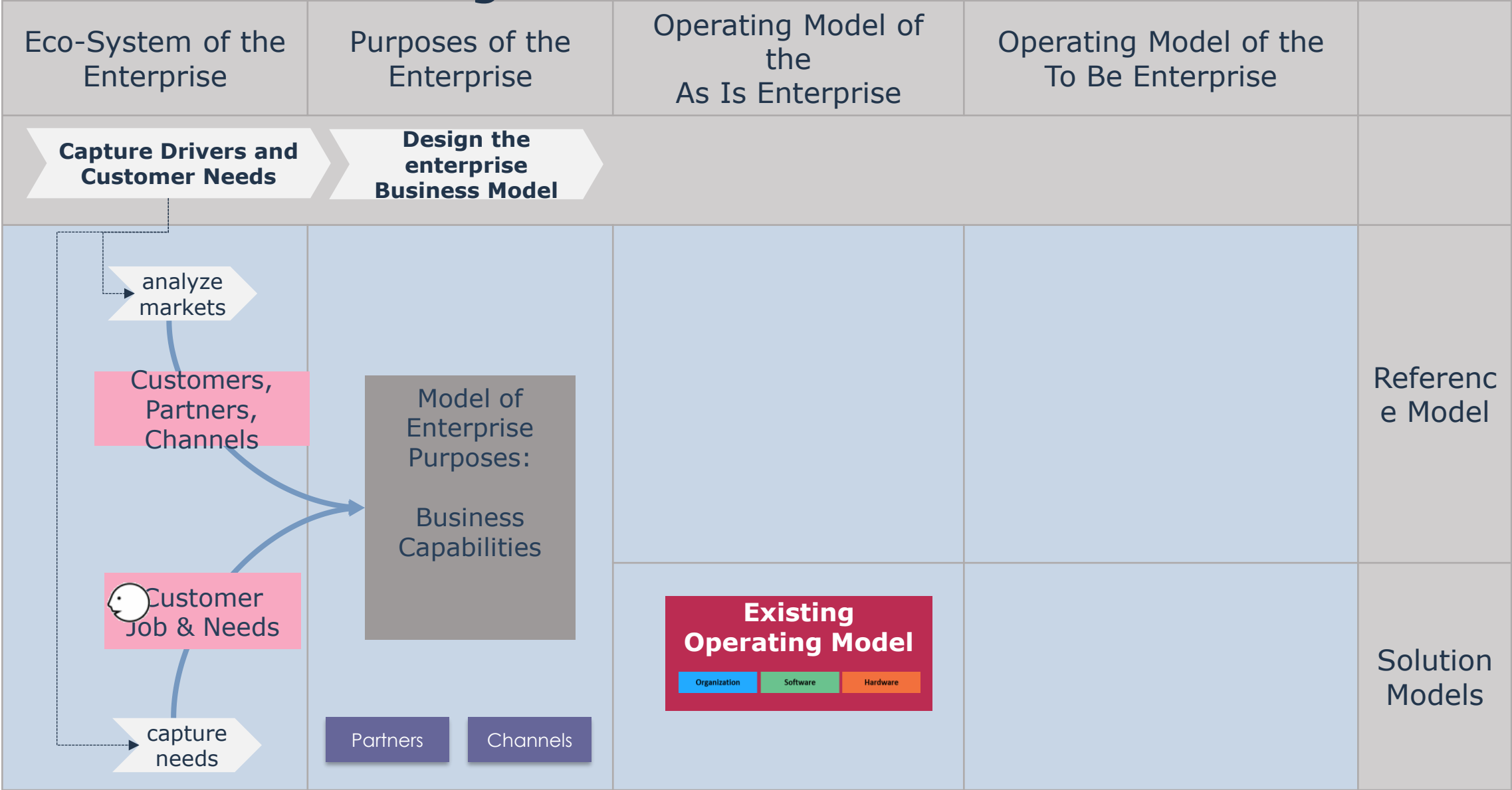
An improved approach is to distinguish Resource-Capabilities from Business-Capabilities.

- Simple but hierarchical & rigid
- Risk of confusing "Business Capabilities" and Resource Capabilities" with "Capability levels"..

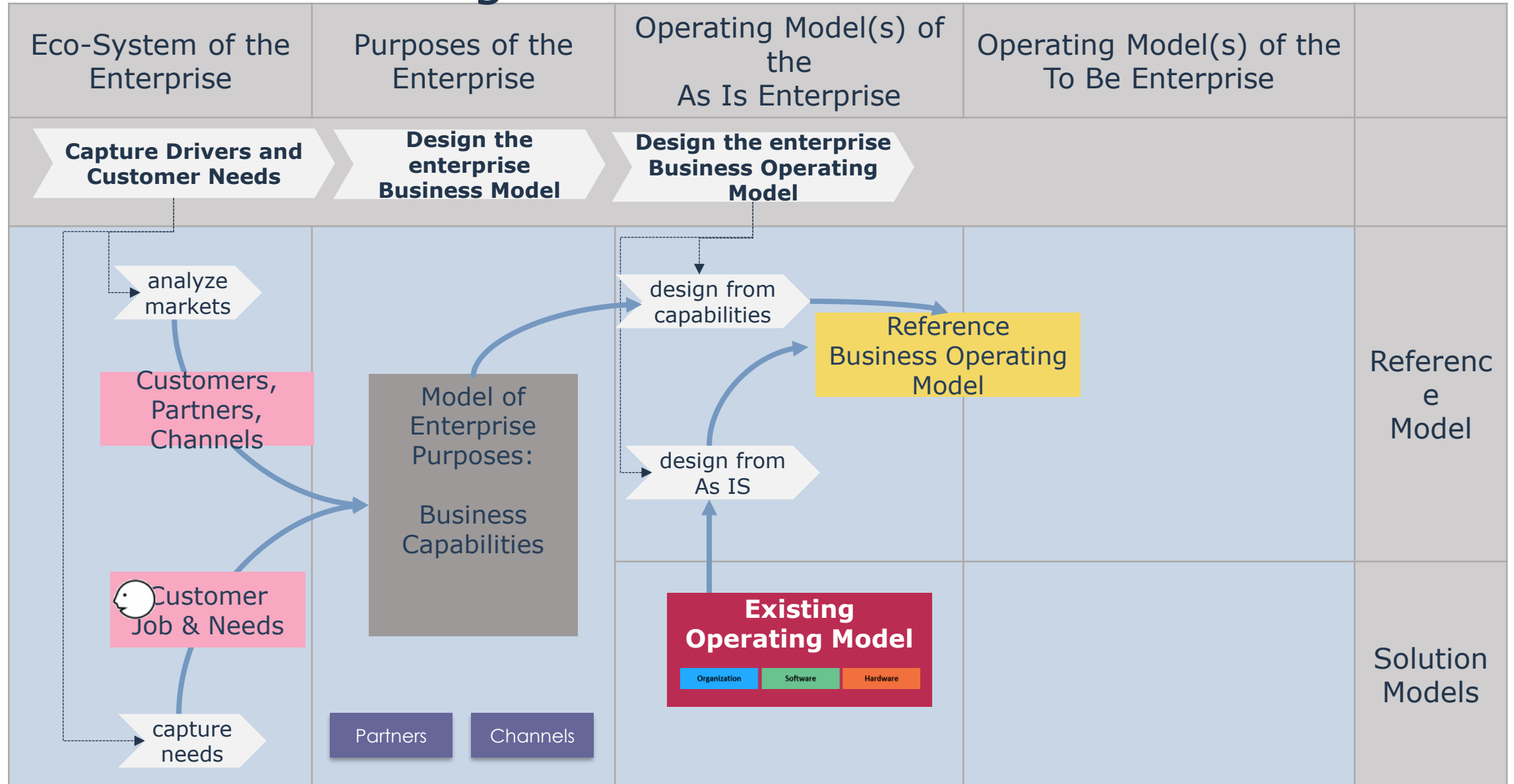
Transformation Design Method Overview



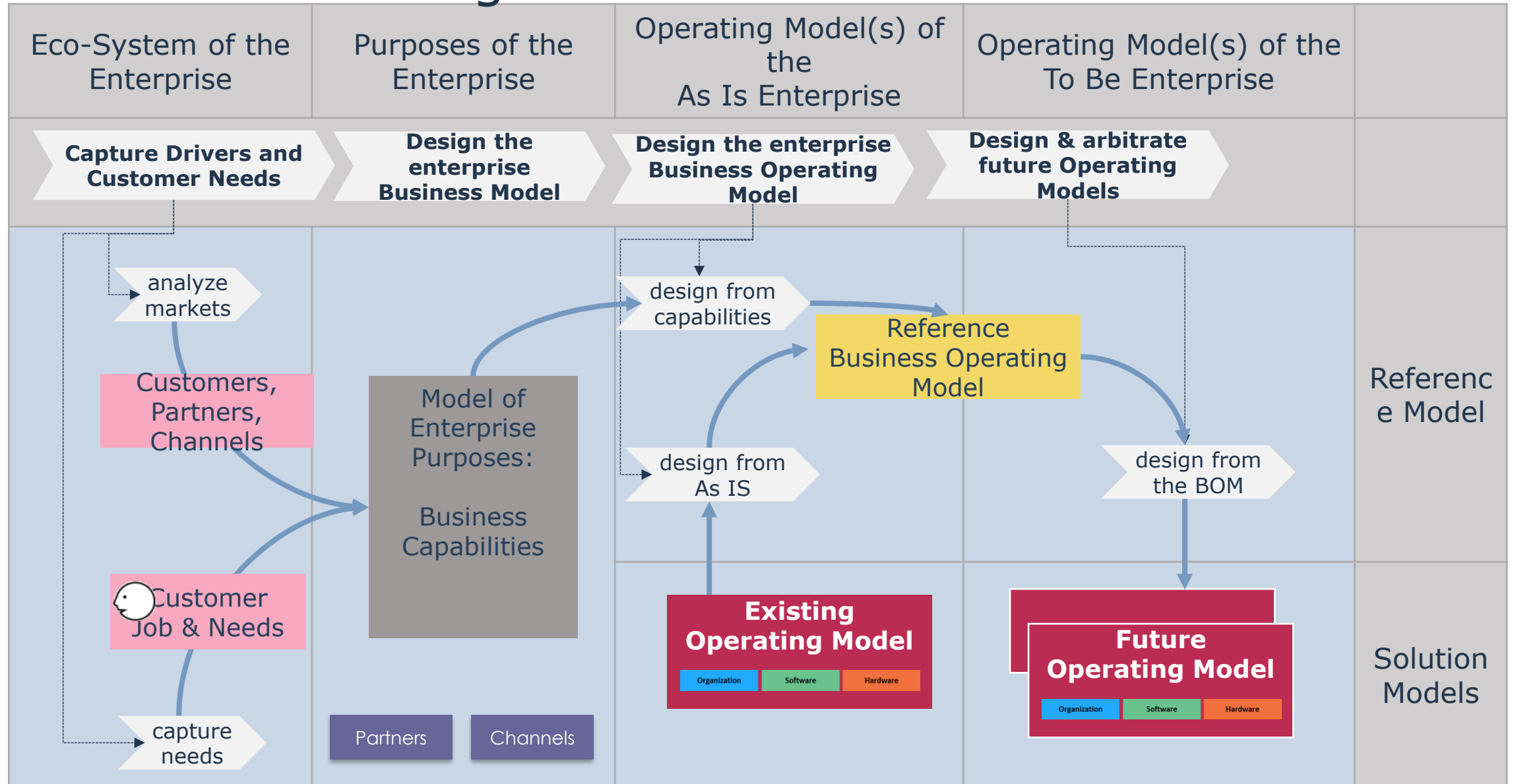
Transformation Design Method Overview



Transformation Design Method Overview

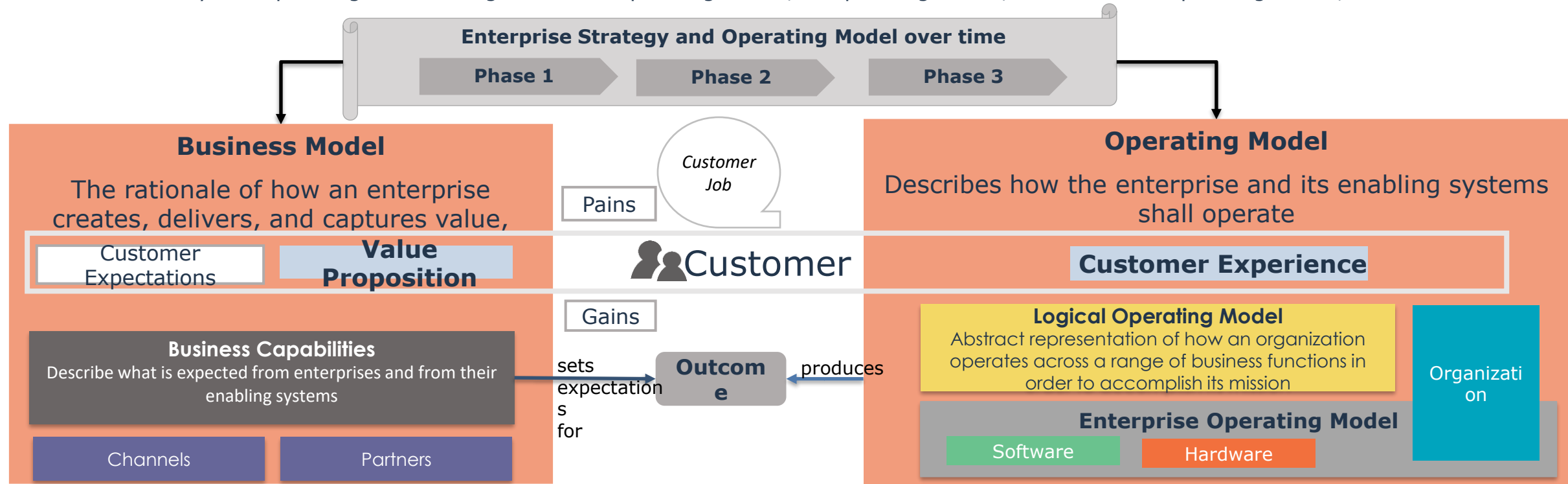


Transformation Design Method Overview



Business Model vs Business Operating Model

- Enterprise Business Model
 - A business model describes the rationale of how an organization creates, delivers, and captures value, in economic, social, cultural or other contexts. The process of business model construction is part of business strategy.
 - In a Capability centric approach, the business model is the planning of business capabilities and their associated value propositions.
- Enterprise Operating Model
 - An operating model is an abstract representation of how an organization operates across a range of domains in order to accomplish its mission. There are two major kinds of operating models:
 - Conceptual business model often called "Business Operating Model".
 - Physical operating models: Organizational operating model, IS operating model, Infrastructure operating model, ...

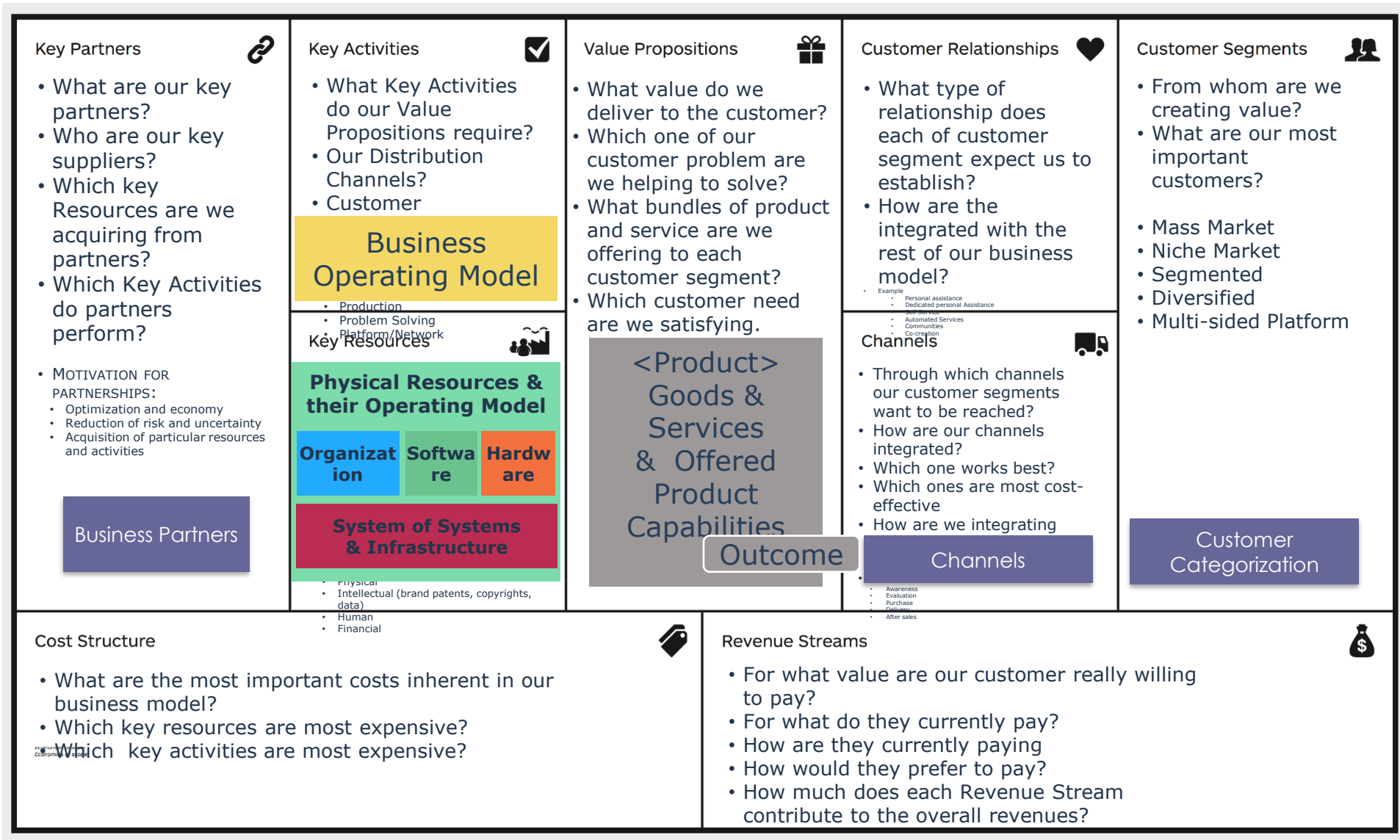


Mapping to management tools

Business Model Canvas

Value Proposition Canvas

Business Model Canvas Mapping



Business Model Canvas – Revenues & Costs Details

Cost Structure  • What are the most important costs inherent in our business model? • Which key resources are most expensive? • Which key activities are most expensive? • How much does each Revenue Stream contribute to the overall revenues?	Revenue Streams  • For what value are our customer really willing to pay? • For what do they currently pay? • How are they currently paying • How would they prefer to pay? • How much does each Revenue Stream contribute to the overall revenues?
---	--

IS YOUR BUSINESS MORE

- Cost driven (leanest cost structure, low price value proposition, maximum automation, extensive outsourcing)
- Value driven (focused on value creation, premium value proposition)

SAMPLE CHARACTERISTICS

- Fixed costs (salaries, rents, utilities)
- Variable costs
- Economies of scale
- Economies of scope

TYPES

- Asset Sale
- Usage fee
- Subscription fees
- Lending/Renting/Leasing
- Licensing
- Brokerage fees
- Advertising

FIXED PRINCING

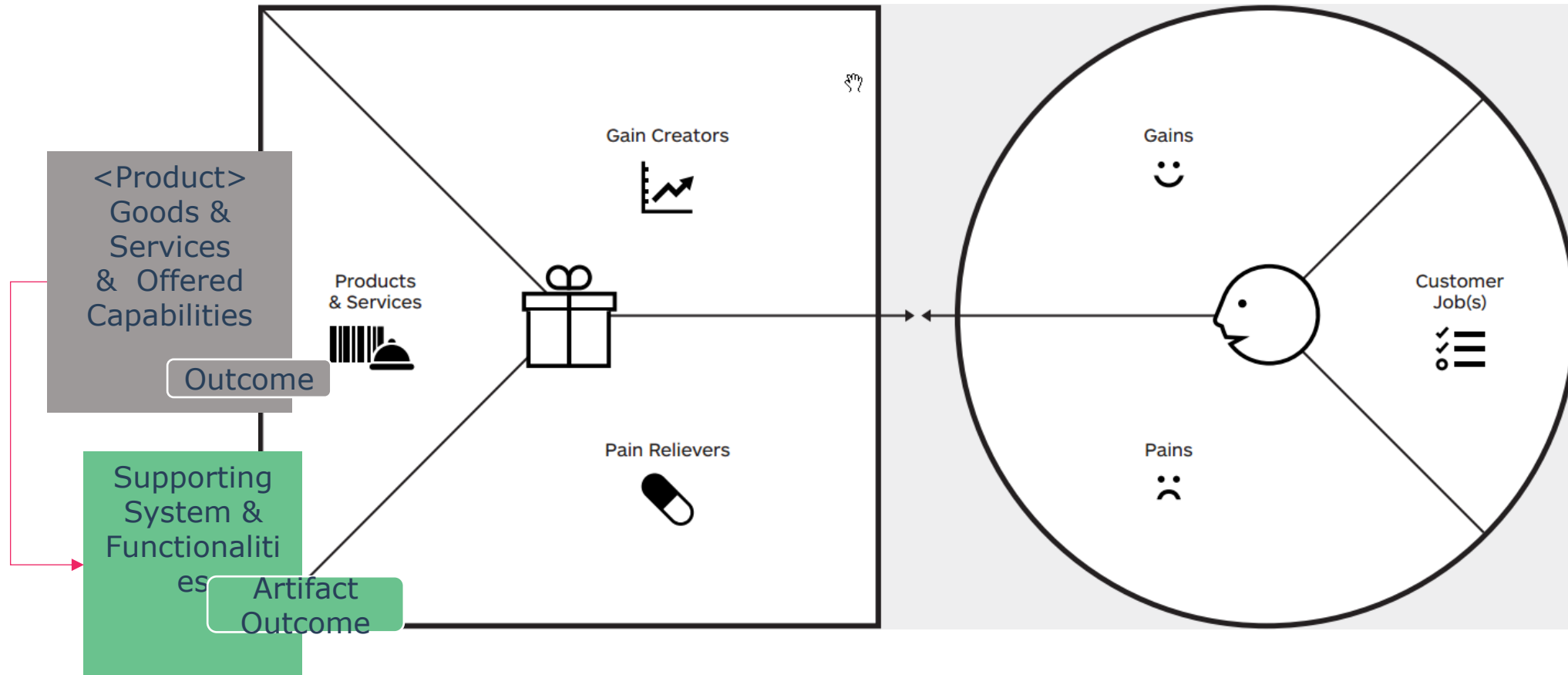
- List price
- Product feature dependent
- Customer segment dependent
- Volume dependent

DYNAMIC PRICING

- Negotiation (bargaining)
- Yield management
- Real-time market

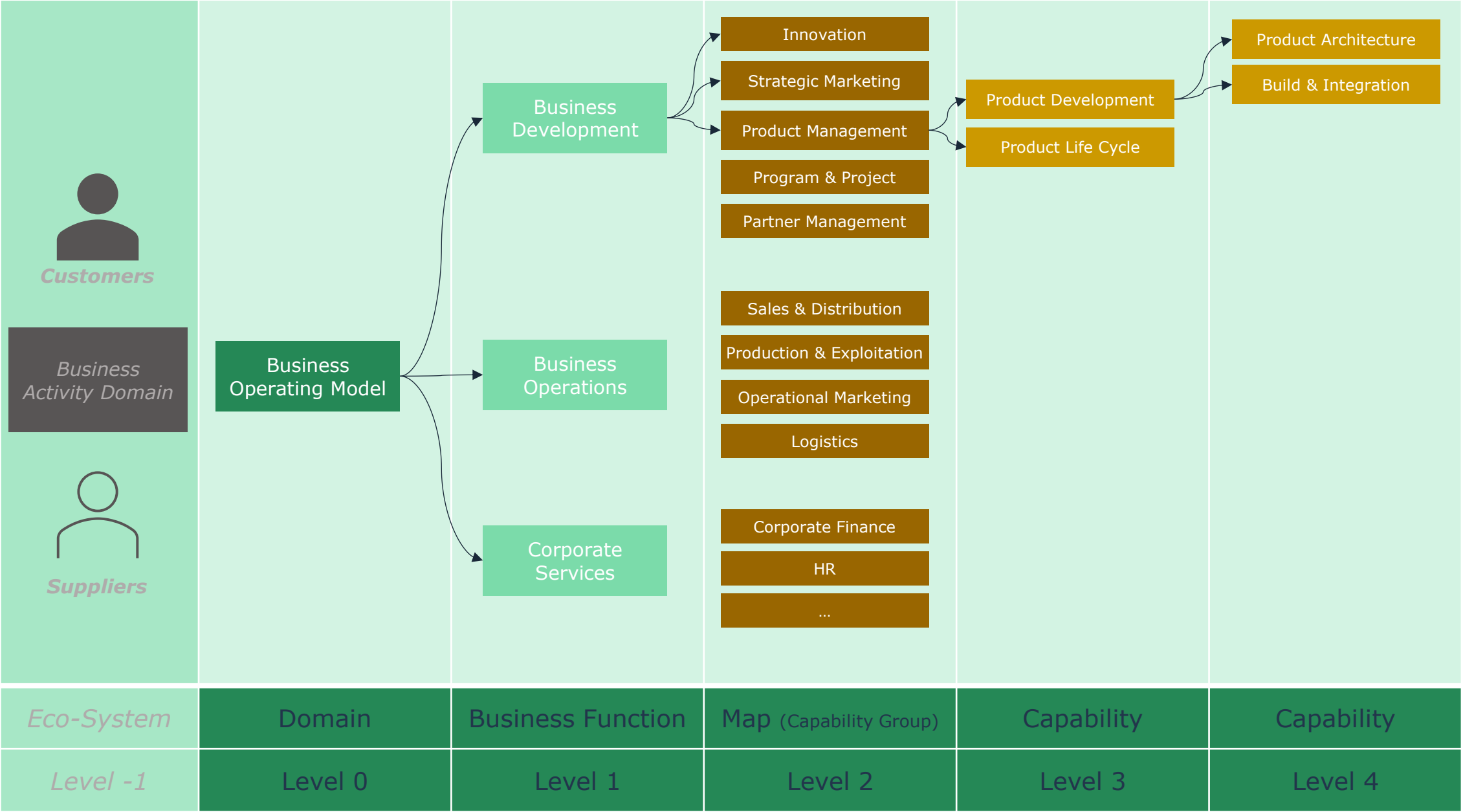
Value Proposition Canvas Mapping

Each feature includes a benefit hypothesis and acceptance criteria,



[Job to be done is included in: HOPEX Metamodel Overview \(demomega.com\)](http://demomega.com)

Enterprise Maps structure – Functional division of labor



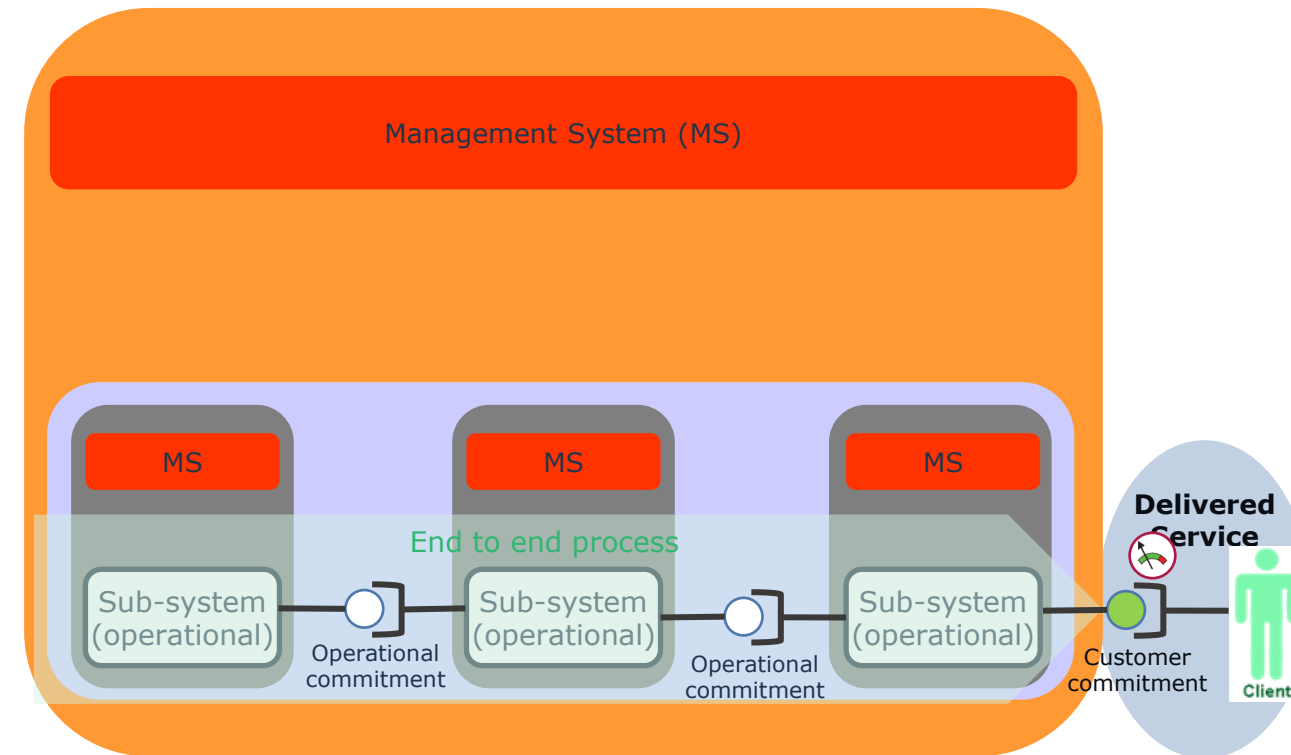


Definition

How are splitted organization ?

- a.by function (e.g., production, marketing, research, finance, and personnel, in the industrial context)
- b.by space (e.g., geography, as territories of sales offices)
- c.by time (e.g., waves of an invading force)
- See Ackoff: [organization.](#)

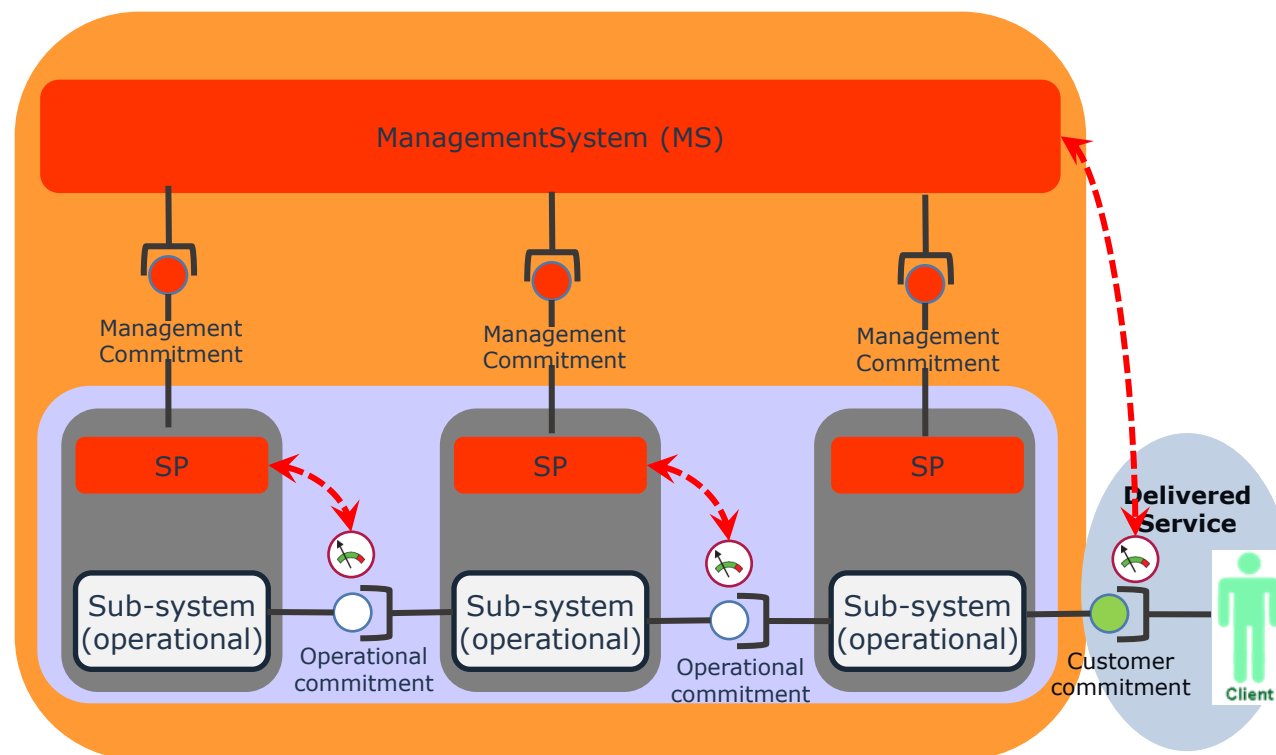
Processes “go across” sub-systems in order to deliver services according to customer commitments.



To be reliable, intermediate added value outcomes must be produced by sub-systems et be manifested by operational commitments

Management of operational commitments

- Each operational commitment performance is managed by the Management System belonging to the operational system in charge



-
- The diagram illustrates a multi-tier service architecture. At the top is a red box labeled "ManagementSystem (MS)". Below it are three identical vertical stacks, each representing a service provider. Each stack consists of a red box labeled "SP" (Service Provider) at the top, a white box labeled "Sub-system (operational)" at the bottom, and a green circle labeled "Operational commitment" in the middle. A solid black line connects the "MS" box to each "SP" box, labeled "Management Commitment". A solid black line connects each "Sub-system (operational)" box to the "Operational commitment" circle, labeled "Operational commitment". A red dashed arrow points from the "MS" box to the "Operational commitment" circle of the first stack. A red dashed arrow points from the "Operational commitment" circle of the first stack to the "Operational commitment" circle of the second stack. A red dashed arrow points from the "Operational commitment" circle of the second stack to the "Operational commitment" circle of the third stack. A red dashed arrow points from the "Operational commitment" circle of the third stack to a green circle labeled "Customer commitment". A green stick figure labeled "Client" is positioned next to the "Customer commitment" circle. A red dashed arrow points from the "Client" to the "Delivered Service" box, which is a red box labeled "Delivered Service".

Three mode of collaboration

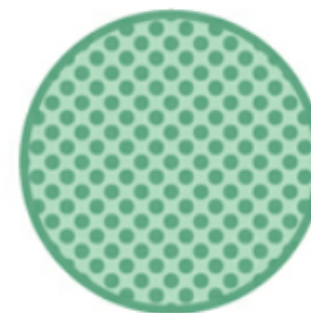
A combination of all three team interaction modes is likely needed for most medium-sized and large enterprises (and these modes are useful to introduce at smaller organizations sooner than many people expect). In addition, one team might use two different interaction modes for two different teams with which it works. We represent these different interaction modes graphically using the patterns in [Figure 7.2](#):



Collaboration



X-as-a-Service



Facilitating

Figure 7.2: The Three Team Interaction Modes

Collaboration mode is shown with diagonal cross-hatching, X-as-a-Service mode is shown with brackets, and facilitating is shown with dots.

Bibliography

- Boulding
- Ackoff
 - [Leadership](#)
 - [TOWARDS A SYSTEM OF SYSTEMS CONCEPTS](#)
- <http://www.utc.fr/~moreauva/Lean2009/systemique.pdf>
- https://en.wikipedia.org/wiki/Mihajlo_D._Mesarovic
- <https://www.maxicours.com/se/cours/les-modes-de-coordination/>
- [Holacracy](#) is a style of enduring organization relying on [purpose](#), delegation and accountabilities.
 - [Holacracy Basics: Understanding Accountabilities | by Chris Cowan, EdD., MDiv. | Holacracy](#)
 - [Enhancing Holacracy-driven environments with the use of OKRs | by Energized.org | Holacracy](#)
- [OKR – Wikipedia](#)
 - This methodology is used to distribute finalities (objectives) across the distribute and modular structure of agile enterprises.
 - [OKR: Learn Google's Goal System with Examples and Templates \(felipecastro.com\)](#)