Business Architecture

Illustrating concepts in these two articles

Service-Oriented Business Architecture

Capability-Oriented Business Architecture

"Very spot on." (Enterprise Architect)
"Very good, thanks👍👍" (Senior manager)
"Excellent." (Director of consulting firm)
"Excellent, thanks for sharing Graham." (Enterprise architect)

Last updated: 23/05/2020

This extract from architect training at http://avancier.website can be found at https://lnkd.in/gAGM_p2

Comments on earlier, shorter, version of this show

- *“… concise material that beautifully connects EA and BA work together.”*
- *“Nicely done.”*
- *“Insightful.”*
- *“I really like it. Well done!”*
- *“Good stuff Graham!”*
- *“This is great. I can use this immediately.”*
- *“Great information. Well done!”*
- *“Great post. Thanks for sharing.”*
- *“Well thought out. Kudos.”*
- *“Oh this is simply brilliant!”*

Roles & concepts in “Skills Framework for the Information Age”

SFIA says **enterprise and business architecture roles** involve:

- interpretation of business goals and drivers
- translation of business strategy and objectives into an "operating model"
- assessment of current capabilities and identification of required changes to them;
- description of **relationships between business system elements**:
  - services [resulting from business activities]
  - processes [sequencing business activities]
  - data/information [created and used by business activities]
  - technologies [supporting and enabling business activities]
  - people [actors playing roles in business activities]
  - organizations [managing people who perform activities]
  - the external environment [notably customers, suppliers, partners, competitors and other stakeholders].

The terminology torture

- Unfortunately, enterprise architects use words drawn from different domains of knowledge, such as
  - the ArchiMate modelling language,
  - business management consulting and
  - software engineering.

- Words like **function, process, capability** and **service** have different meanings in different domains.
  - In many published examples, Functions are called Capabilities
  - In the BIAN hierarchy, Functions are called Services
  - In the APQC PCF hierarchy, Functions are called Processes
  - Some think Functions are Organization Units

Business systems as I/O transformers

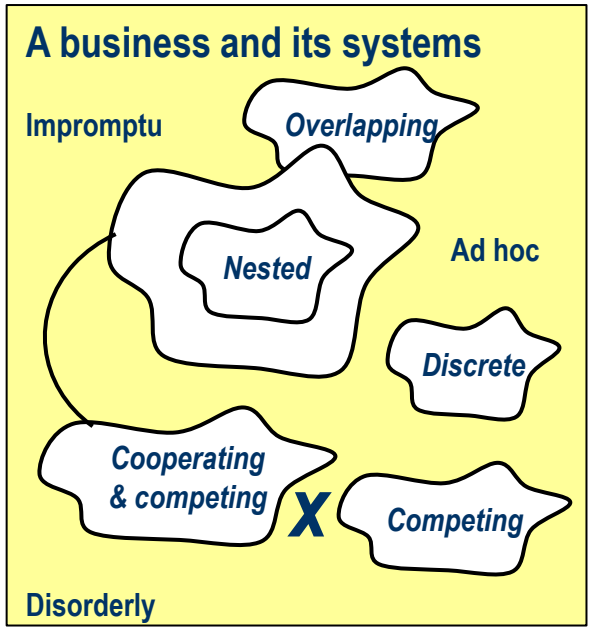
- It is said that EA regards a business as a system of systems.

- Seen from the outside
- Most business systems are “open” and transform **inputs** into **outputs**

- General system theory embraces flows of energy and forces
- EA focuses on flows of materials (sometimes) and information (mostly)

Focusing on the system of interest

- You can find countless different systems in a business
  - large and small
  - nested and overlapping
  - more and less connected by flows
  - synchronized and out of step,
  - cooperative and competitive.



- At any one time, architects focus on what is called "the system of interest", or a relationship between systems.

Two kinds of system boundary

- To scope a system of interest, you may draw
- a boundary around a physical entity
  - such as a farm, a factory, a shipyard, or some part thereof.

- a legal or logical boundary around interacting actors
  - distributed in space and connected by information flows.

Inside a system

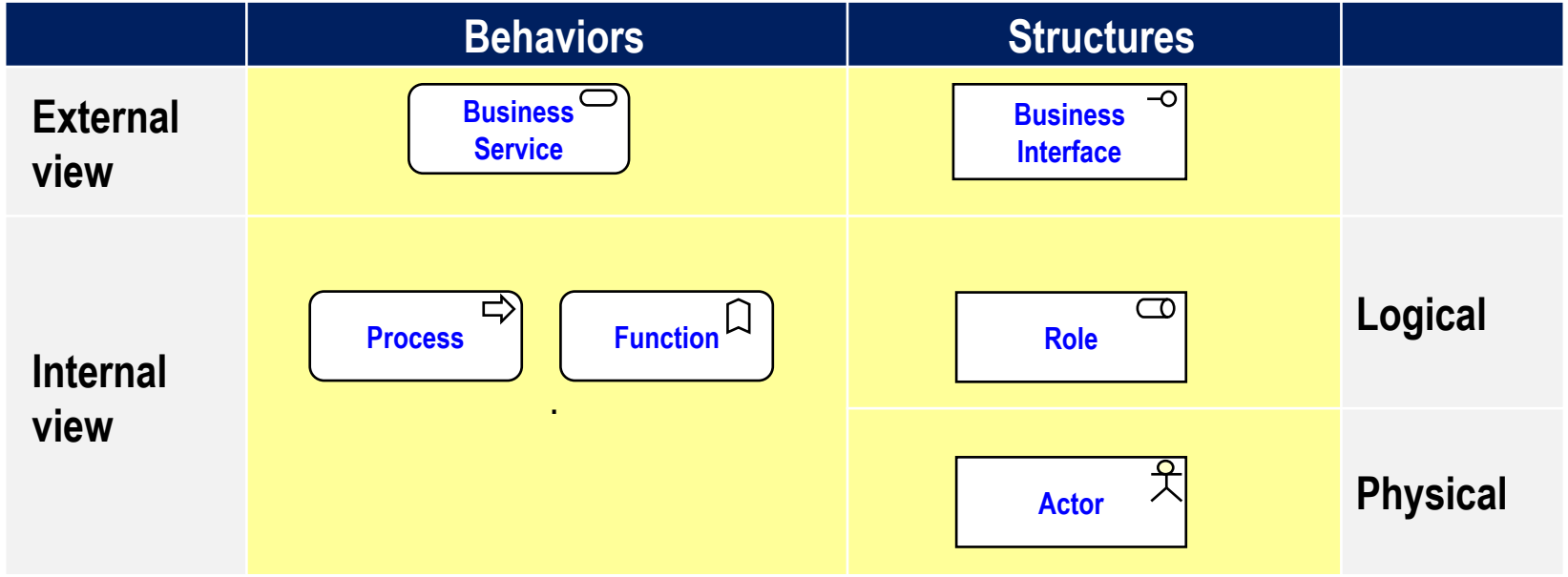
- **Actors** (or components) interact in
- **Activities** (or processes) to produce outputs and so meet
- **Aims** (or goals) of value to external actors and system sponsors

- An actor is a structure (or continuant) that can perform activities
  - Individual actors and components are bought, hired or employed

- An activity is a behavior (or occurrent) that changes or makes something
  - Atomic activities are performable without further definition.

In ArchiMate's generic model of a business system

- *Behaviors* are performed by *active structure elements*.
  - the active structures are *actors* who play *roles*
  - the behavioral elements are called *services, processes* and *functions.*



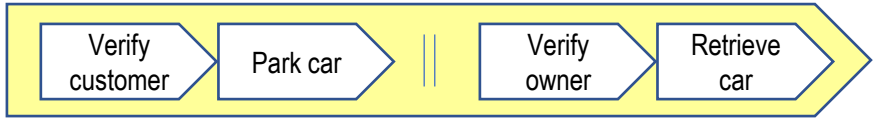
Service-oriented architecture

- Business systems analysis and design typically proceeds by defining elements as follows.
    1. **Goals** or outcomes of interest to external actors and sponsors
    2. **Services** (entry and exit conditions) external actors need to reach goals and outcomes
    3. **Processes** (sequences of activities) to complete or deliver services.
    4. **Roles, data** and other structural resources needed to perform processes
    5. **Actors** and organizations to perform roles and manage resources.

- The sequence is flexible.
- E.g. The choice between robot or human actors might lead you to modify the roles or the processes.

An example of Business Architecture Q&A

1. Why? What are the goals of the business?
   - Attract more customers to the hotel.
2. What services will the business provide to those ends?
   - Free valet parking of a car (along with other services)
3. What processes must be performed to deliver those services?



4. What roles will perform activities in processes?
   - Valet (3 actors)
5. What data entities do activities need?
   - Etc.
6. What locations will actors work at?
   - Hotel entrance and car parks
7. What organization units will manage the actors?
   - Front desk management

Relationships between business system elements in SFIA

Motivations

A **goal/objective** is an outcome to be achieved, declared in response to **drivers**

Behaviors meet motivations.

A **service** delivers a result that contributes to some goal(s).

A **process** sequences some or all activities needed to complete a service.

An activity within a longer process may be assigned to a function, org unit or role, and mapped to data entities and app services

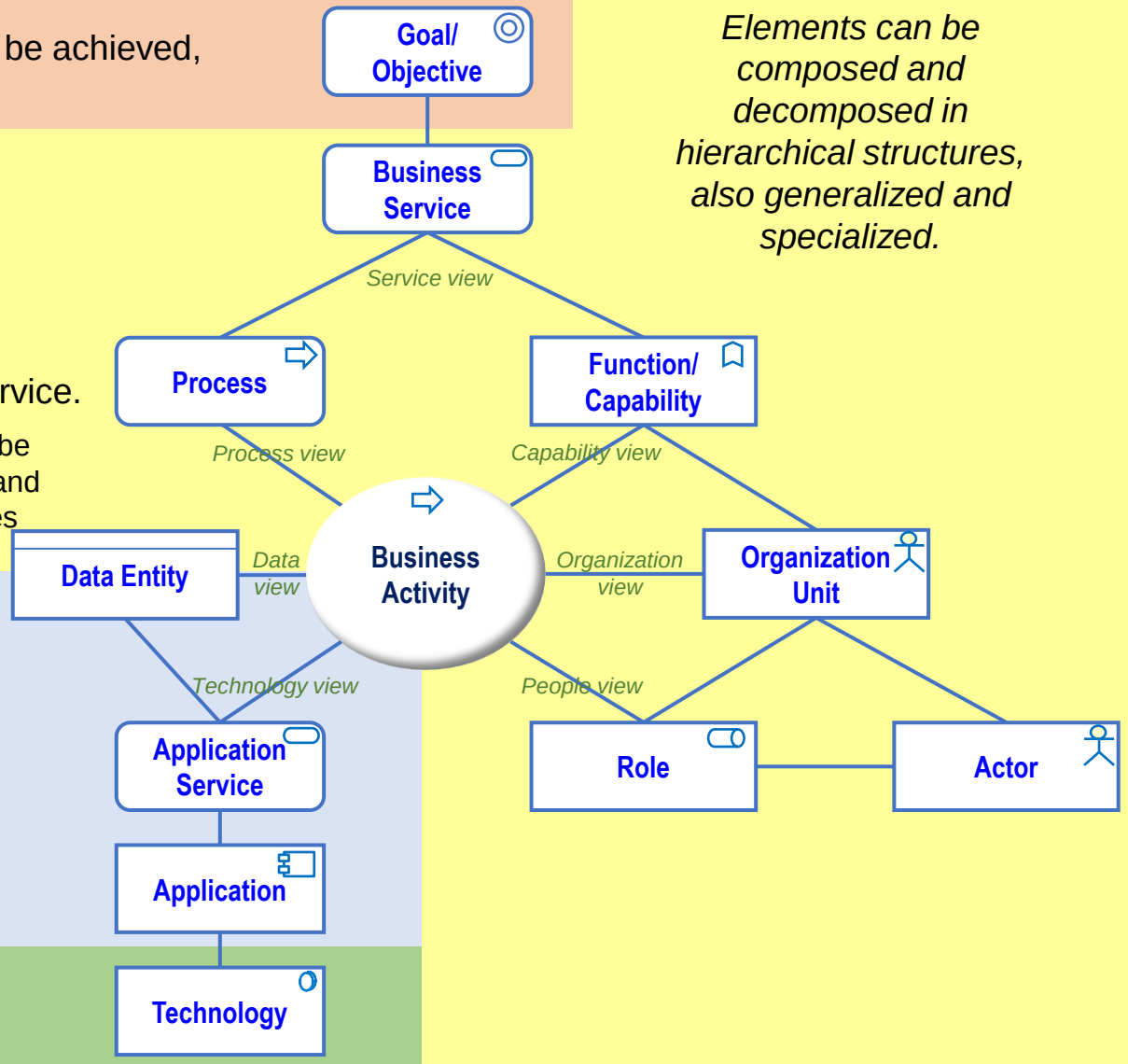
Information is created and used

A **data entity** records something a business must remember.

An **application service** enables business activities

An **application** provides application services

A **technology** provides computing services to enable applications



Elements can be composed and decomposed in hierarchical structures, also generalized and specialized.

Logical before physical

To scope and discuss a large business we divide it into logical divisions called functions or capabilities

A **function** is a logical division of business activity (grouping lower level functions or atomic activities)

A **capability** is a logical division of the ability to function (grouping lower level capabilities or resources)

Structures perform behaviors.

An **organization** unit groups lower level units or roles a manager can manage.

A **role** groups activities an individual actor can be asked to perform

An **actor** is an individual that performs one or more roles.

Simplified business architecture method

1. **Goals** or outcomes of interest to external actors and sponsors
2. **Services** (entry and exit conditions) external actors need to reach goals and outcomes
3. **Processes** (sequences of activities) to complete or deliver services.
4. **Roles, data** and other structural resources needed to perform processes
5. **Actors** and organizations to perform roles and manage resources.

Goals and other precursors to architecture definition

- Architects start by looking at a system from the outside
  - analyse the system's environment
  - identify stakeholders and their concerns
  - identify what a system should achieve (its goals) and
  - perhaps what it currently achieves in operation (its outcomes).

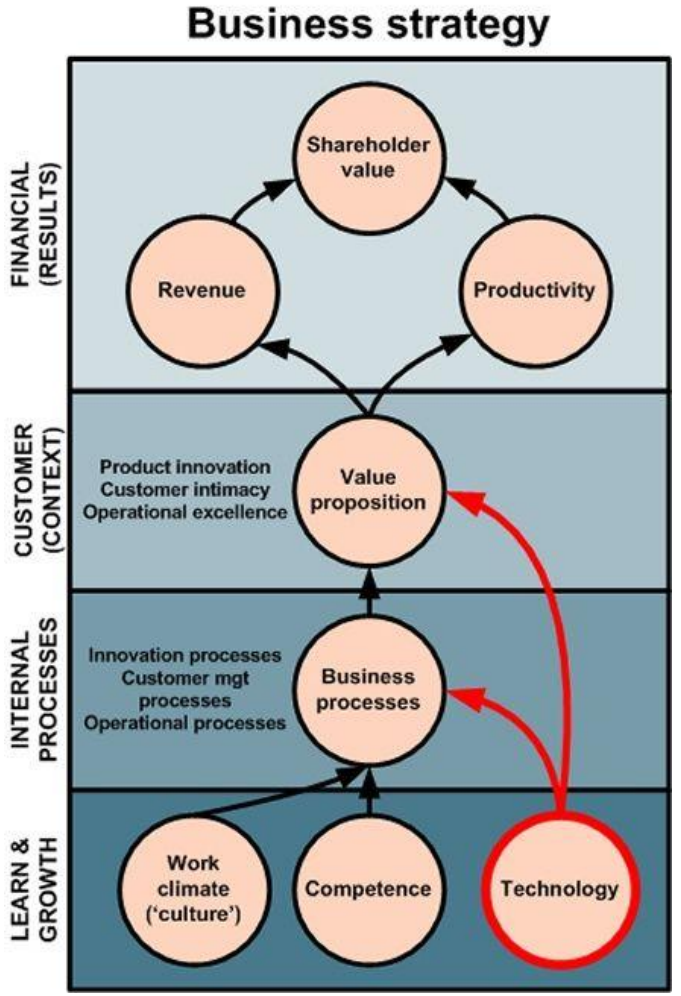
- They must also consider risks and constraints
  - time, budget, resources, legislation etc.

Business architecture usually defined by managers

Goal/objective (aim) structure

Goal/ Objective

- Typically decomposed from the top down
- May be spread across a “balanced score card”
- May be aligned with the organization structure
- Should be SMART, with quality measures (variables)



This extract from architect training at http://avancier.website
can be found at https://lnkd.in/gAGM_p2

ArchiMate and TOGAF

- ArchiMate speaks of *goals* and *outcomes*.
- TOGAF speaks of *goals, objectives* and *architecture requirements*.

- Some goals are more functional.
  - E.g. A retailer wants to fill an identified gap in the market.
  - A tank must be designed to traverse rough terrains.
- Some goals are more non-functional.
  - E.g. Double sales volume this year.
  - Make a bigger profit.
  - Resolve 90% of complaints to the satisfaction of customers.
- Some goals combine functionality and non-functional qualities.
  - E.g. an army wants to put a thousand boots on the ground anywhere within 24 hours.

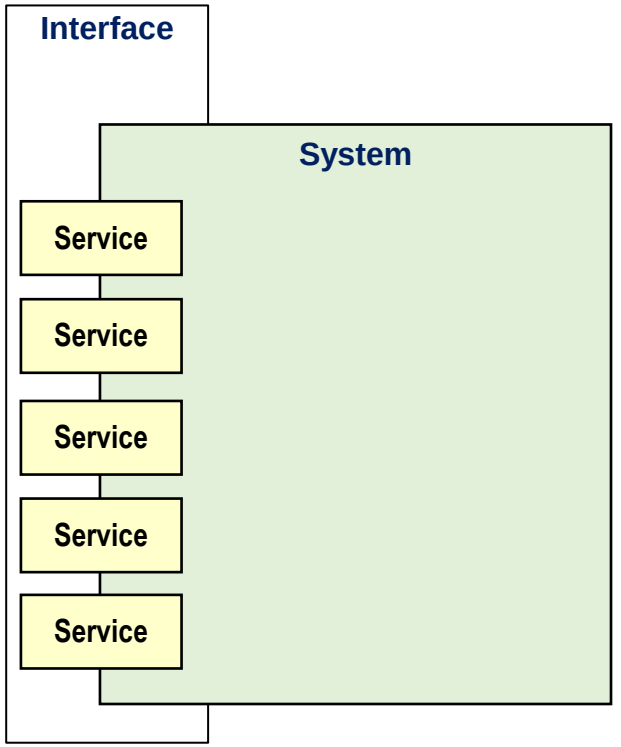
Simplified business architecture method

1. **Goals** or outcomes of interest to external actors and sponsors
2. **Services** (entry and exit conditions) external actors need to reach goals and outcomes
3. **Processes** (sequences of activities) to complete or deliver services.
4. **Roles, data** and other structural resources needed to perform processes
5. **Actors** and organizations to perform roles and manage resources.

Mapping services to goals

- Architecture frameworks like IAF and TOGAF
- and modelling languages like ArchiMate
- distinguish external and internal views of a system

- They hide the internal structures and behaviors of systems and subsystems behind interfaces.
- Architecture definition starts with services that actors require from the system, to meet declared aims.



Services

- **Services** are things a system or subsystem does for external actors.
- They are discrete event-driven behaviors that terminate in results of value to customers, clients or users.
- They can be coarse-grained (e.g. build a house) or fine-grained (e.g. book an appointment).

How do we document services?

- First name the service

Barber Services
- Hair cut – £20
- Shave - £5
- Manicure - £10

Logistics Services
- Delivery
- Express delivery
- Recorded delivery

- Clustering related services under headings (e.g. cleaning services) may correspond to a function

AutoXpress Services
- Fit tyres
- Check-up and oil change
- Full annual service
- Check brakes
- Repair brakes
- Check exhaust
- Replace exhaust
- Inspect battery
- Replace battery
- Align wheels
- Replace windscreen wipers
- Fit bulbs
- Replace shock absorbers

A service contract contains

- Name (e.g. build a house, or book an appointment).
- Entry conditions (inputs and other preconditions)
- Exit conditions (outputs and other post conditions, inc, perhaps "value delivered")
- Qualities of service (speed, volume and other measures, inc. perhaps price).

Detailing services in contracts

In TOGAF, architecture requirements include business and application service contracts.

Service name
Entry conditions
- Inputs/Supplies
- other Preconditions

Exit conditions
- Outputs/Products
- other Postconditions
- The value to users of the service

Quality of Service measures
- Time
- Volume
- etc.

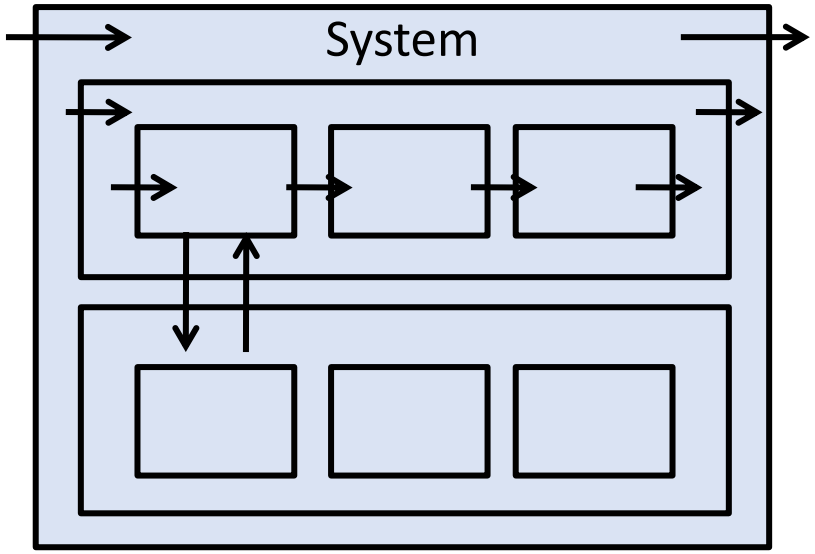
Service: **Park guest's car**
Entry conditions
- Input: Car keys, Room number
- Precondition: Check in complete

Exit conditions
- Output: Car parked
- Postcondition: Car location recorded

Quality of Service measures
- Time: 5 minutes
- Volume: 150 a day
- etc.

Systems can be nested

- In ArchiMate and TOGAF, the external behavior of a system is defined by the services it provides to actors its environment.
- Where a system is divided into subsystems (aka components or building blocks), each is encapsulated by an external/internal interface.
- So, to complete a coarse-grained service at the highest level of system definition may require many subsystems to provide finer-grained services.



System performance - qualities of service

- A system's performance characteristics are primarily specified as "qualities of service" in service contracts. E.g.
  - speed,
  - volume,
  - availability,
  - security,
  - scalability,
  - usability
  - Integrity
  - price and cost.

Simpler specification

- The system specification may be simplified by rolling up some service qualities to the system level.
- E.g. all the many services offered by one system are available for the same hours each day.

- The qualities can be measured at run-time against what is declared in service contracts.

Simplified business architecture method

1. **Goals** or outcomes of interest to external actors and sponsors
2. **Services** (entry and exit conditions) external actors need to reach goals and outcomes
3. **Processes** (sequences of activities) to complete or deliver services.
4. **Roles, data** and other structural resources needed to perform processes
5. **Actors** and organizations to perform roles and manage resources.

Process

- Internally, a system can be defined in terms of activities performed by actors - to complete required services.

- "A business process represents a *sequence* of business behaviors that achieves a specific *result*.... ArchiMate 3.1

- A process *sequences* what can or should be done – it sequences activities that lead to a result of value to one or more actors. E.g.
  - Advertise a product
  - Accept a payment
  - Receive and stock a product
  - Deliver a product to a customer
  - Bid for a specialist contract.

An end-to-end process or Value Stream (IT4IT)

Stages

Activities in each stage

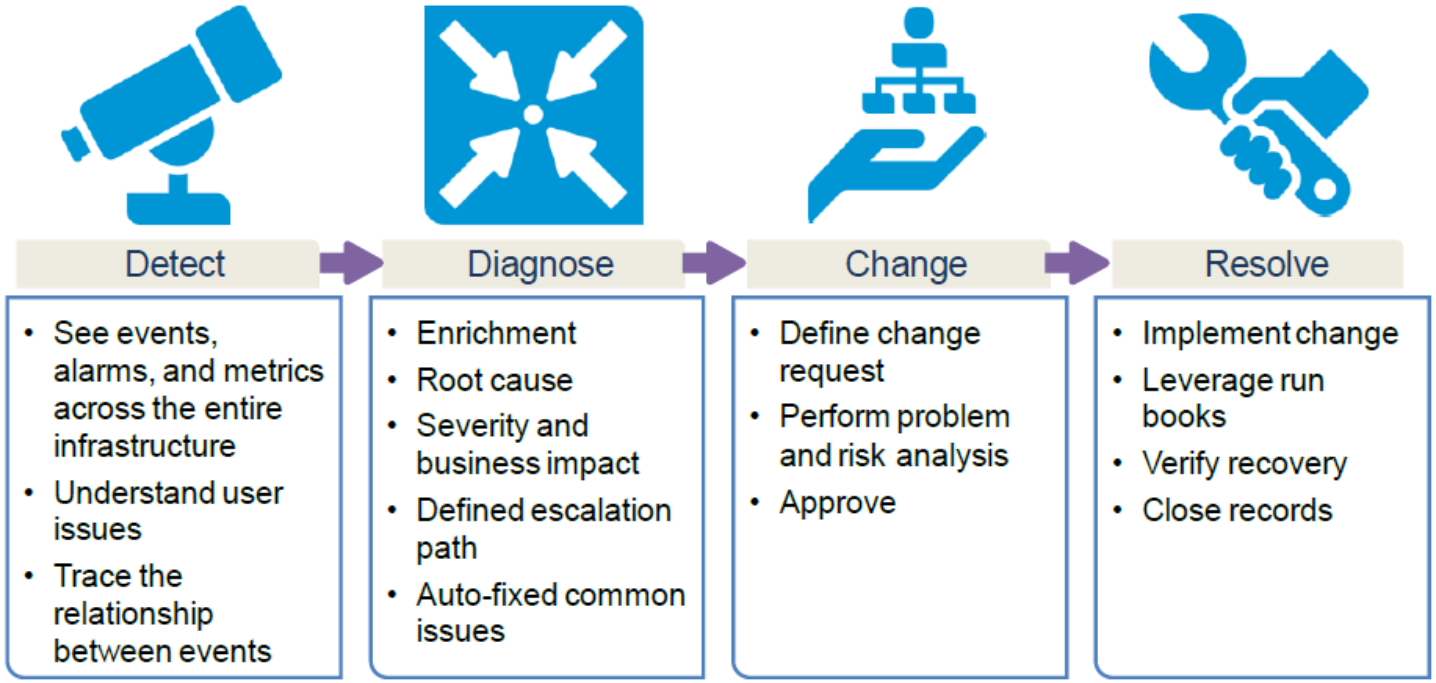
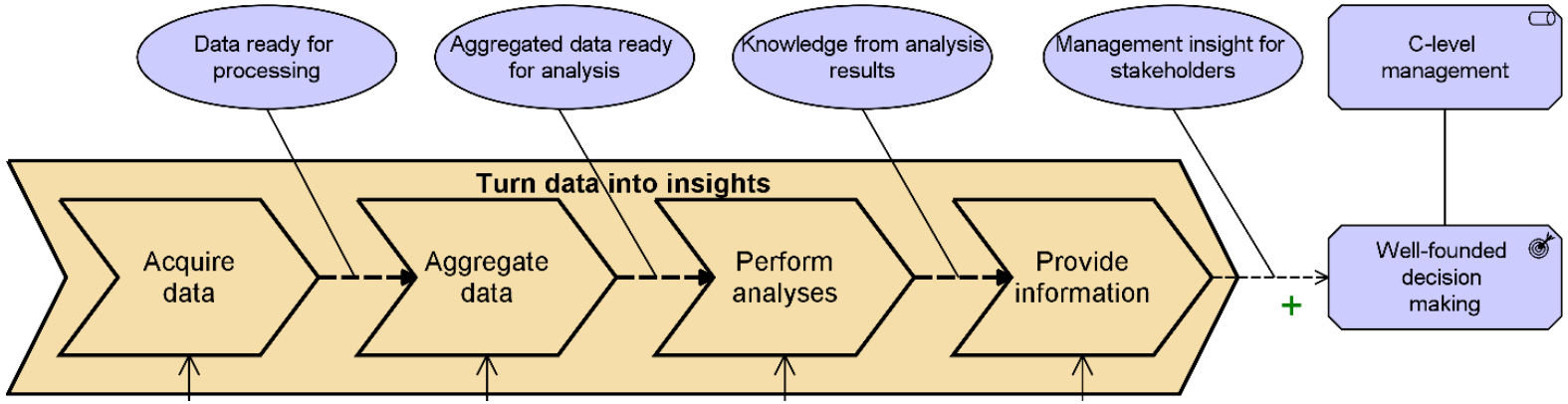


Figure 7: Detect to Correct Activities

A Value Stream differs from LEAN's **Value Stream Map**, where a goal is to eliminate waste from material processing.
A Value Stream that delivers successive Business Services to customers may be called a **Customer Journey.**

Process decomposition

A value stream is a high-level, “end-to-end” **process,** divided into broad stages each divisible into several/many activities.



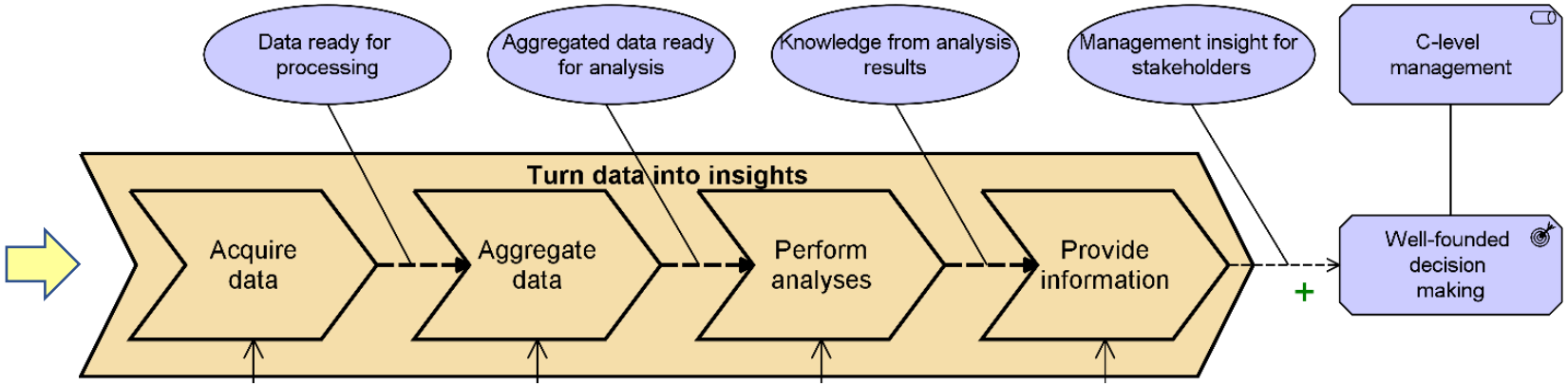
The process can be be divided into parallel processes and decomposed into shorter processes.

It may be decomposed through scenarios and detailed process flows to application services/use cases.

Mapping processes to services

- A service contract encapsulates an end-to-end process (aka value stream) by which a system (if successful) proceeds from the service's entry conditions to its exit conditions.

Service: **Turn data into insights**
Entry conditions
Input: Data
Precondition: Data is available
Exit conditions
Output: Insight given to managers
Value: Better decisions made
Quality of Service measures
Time: I day
Volume: Once a week
etc.



Mapping processes to services

A service contract *encapsulates* internal behaviors

	Behavioral view	Structural view
External view	**Service contracts**	Interface definitions
Internal view	**Processes, Value Streams**	

"A business process represents a *sequence* of business behaviors that achieves a specific *result*.... ArchiMate 3.1

“A value stream represents a *sequence* of activities that create an overall *result* for a customer, stakeholder, or end user.” ArchiMate 3.1

Beware

- In line with ArchiMate and TOGAF standards, here are the first four dictionary definitions of *process* I found:
  - a series of actions or steps taken in order to achieve a particular end.
  - a series of progressive and interdependent steps by which an end is reached.
  - a series of actions which are carried out in order to achieve a particular result.
  - a sequence of interdependent and linked procedures which, at every stage, consume one or more resources.

- However, some business management/architecture gurus use the term process to mean what is called a function in the EA tradition.

Mapping functions to processes

- Processes and functions are orthogonal views of the same activities.
  - A process *sequences* what can or should be done
  - A function names what can or should be done – a group of cohesive activities.

- In the ArchiMate standard: "A business function represents a collection of business behavior based on a chosen set of criteria." E.g.
  - Advertising goods.
  - Accepting payments.
  - Handling goods-in.
  - Organizing logistics.
  - Bidding for contracts.

A Functional Decomposition Diagram

- Functions can be thought of as logical business components.
- They can be nested in a composition hierarchy, from large down to small.
- The hierarchy can be used to
  - scope work to be done (heat maps)
  - classify other system elements

- Functions can instead overlap in terms of the activities they group, or be defined as stand alone.

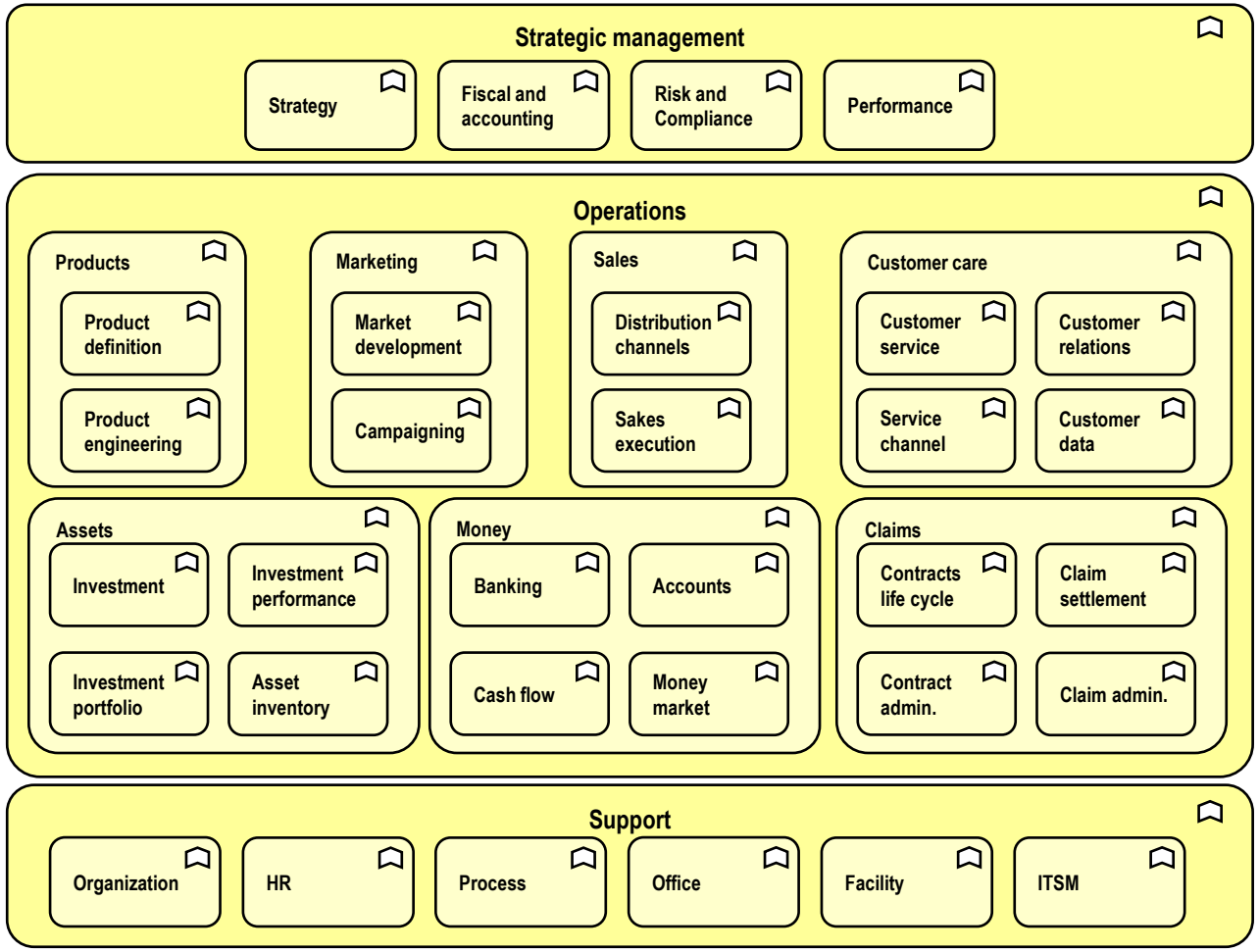


Diagram adapted from an ArchiMate example

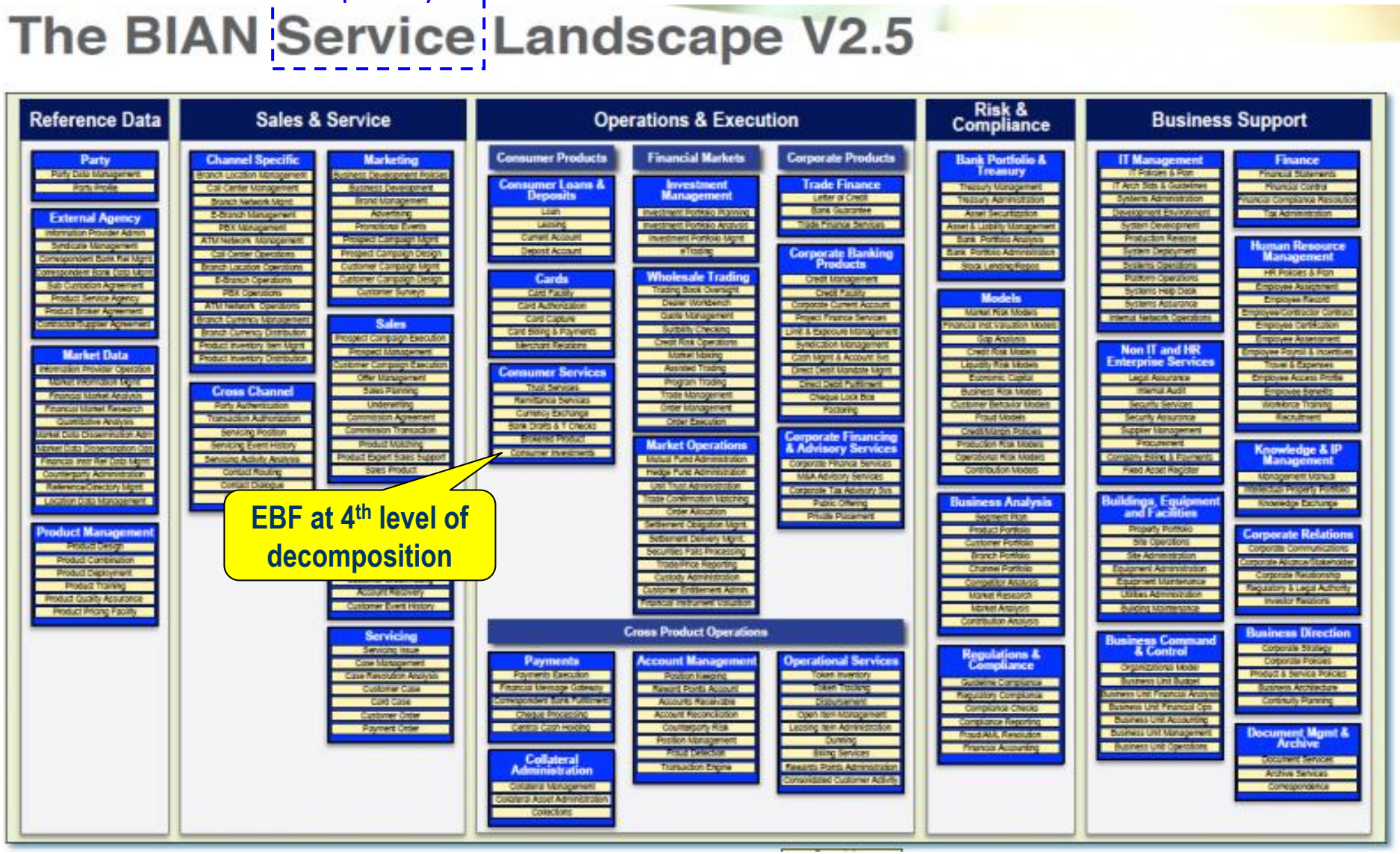
Generic building blocks

You may find a logical hierarchy in the form of *generic reference model* for a business of your kind - then tailor it to fit your business.

TOGAF catalogs the Services provided by each Function.

Defined thus, a Function might be a candidate for outsourcing.

Better, Function or Capability!



This extract from architect training at http://avancier.website
can be found at https://lnkd.in/gAGM_p2

What cohesion criteria should we use

- How to cluster activities into lower-level functions into higher functions, to build a hierarchy?
- Should they
  - meet the same goal?
  - create the same data?
  - need the same resources?

- Whatever you choose to
  - scope work to be done
  - categorize related business architecture elements

Common functions

- In functional decomposition theory, no function or activity is duplicated under different branches of the hierarchy.

- To achieve that you must stop when you reach a common function and document it separately (or in a "common functions" leg of the hierarchy).

Mapping functions to processes

- A function might correspond to one process.
- However, since both functions and processes can be decomposed, they may be coarse or fine-grained, and their relationship is many-to-many.

- One broad function could encapsulate all the activities in several short processes.

- One end-to-end process may coordinate activities in several narrow functions
  - which may each be mapped to activities sequenced in that process

One broad function could encapsulate all the activities in several short processes.

- Example to be added

One end-to-end process may coordinate activities in several narrow functions

Three building blocks <serve> this process stage

Serve means <provide functionality to>.

These “building blocks” could be **functions** or **organization units** or (here) **capabilities.**

Such a coarse-grained process may be decomposed into shorter processes with shorter steps, and at that lower level you may link steps to **roles** played by individual actors.

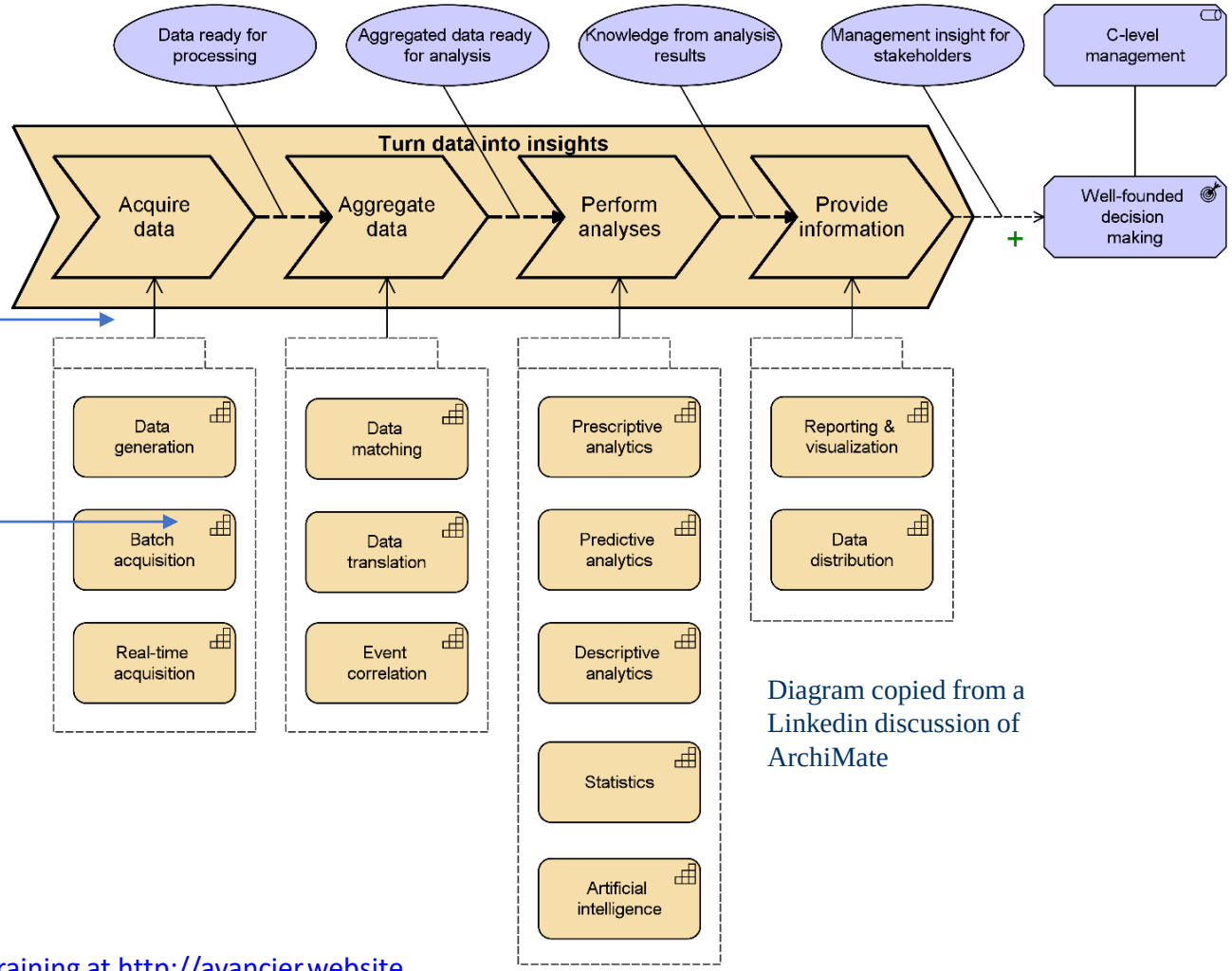
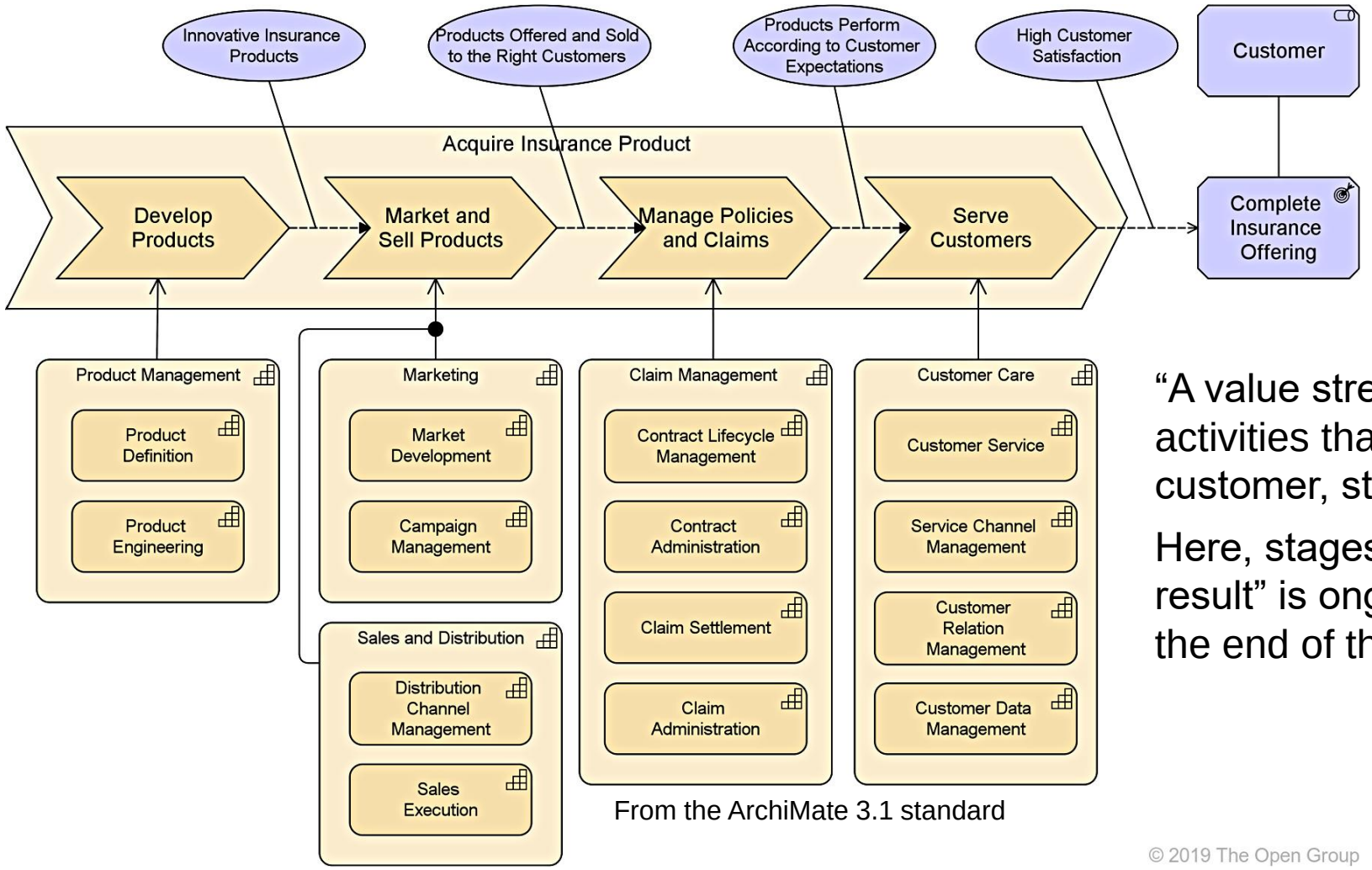


Diagram copied from a Linkedin discussion of ArchiMate

This extract from architect training at http://avancier.website
can be found at https://lnkd.in/gAGM_p2

Value stream diagrams that aren't really processes



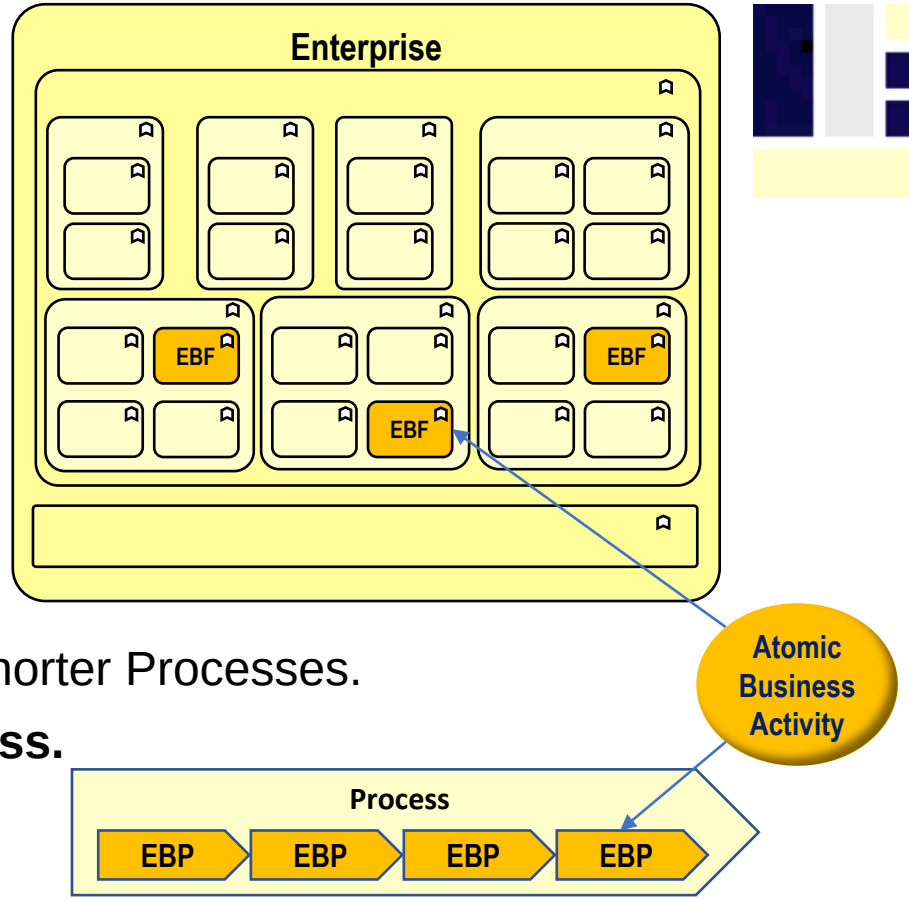
From the ArchiMate 3.1 standard

"A value stream represents a sequence of activities that create an overall result for a customer, stakeholder, or end user."

Here, stages run in parallel and the "overall result" is ongoing rather than delivered at the end of the process.

can be found at https://lnkd.in/gAGM_p2

A verification principle



A Function is a logically cohesive group of activities

- At the bottom of a FDD is an **Elementary Business Function.**

A Value Stream or end-to-end Process can be decomposed into shorter Processes.

- A bottom-level Process step is an **Elementary Business Process.**

Structural decomposition normally stops higher than behavioral decomposition, but there is a verification principle.

- Every EBP must either appear as an EBF in the FDD - meaning the "atomic activities“ are the same,
- or else be assignable to an EBF - if the Functions are not decomposed as far as the Processes.

Simplified business architecture method

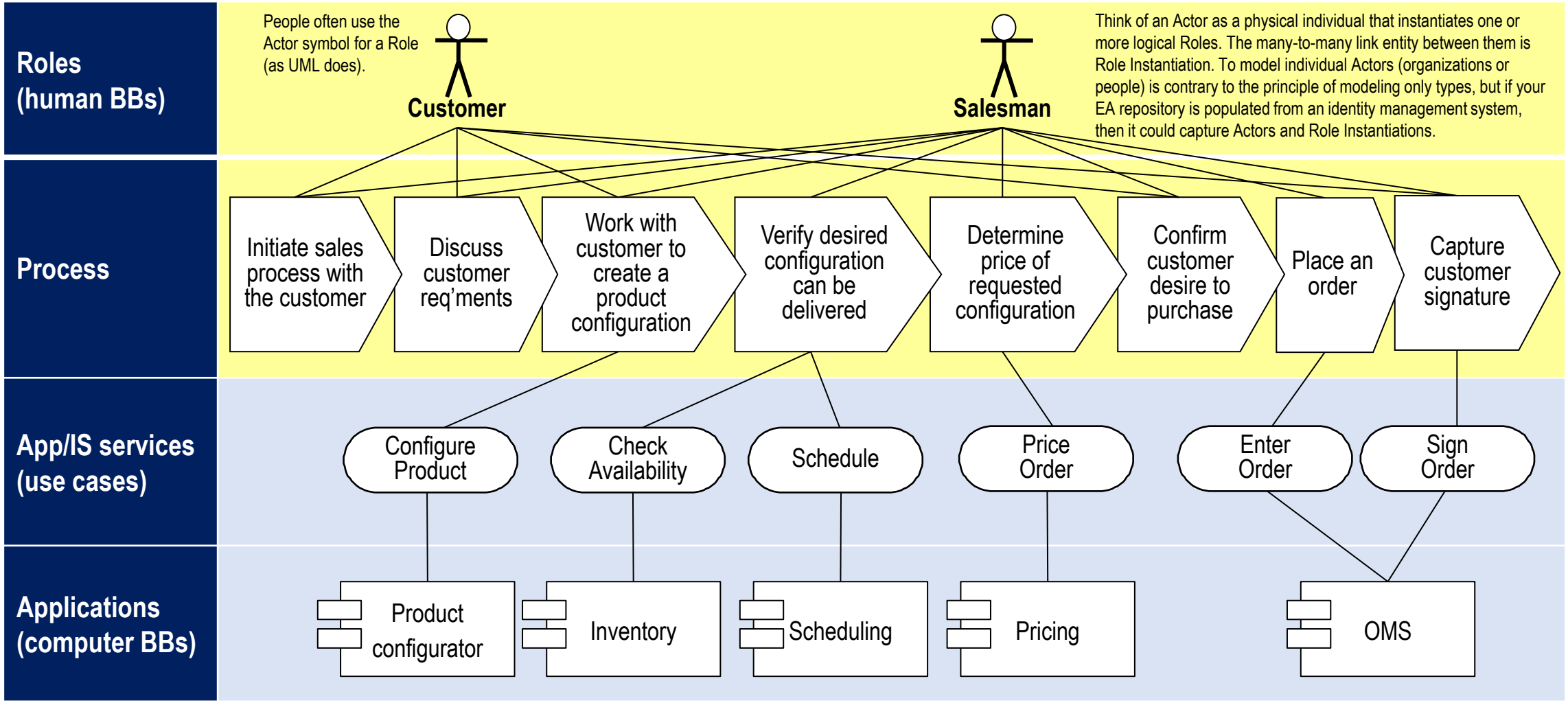
1. **Goals** or outcomes of interest to external actors and sponsors
2. **Services** (entry and exit conditions) external actors need to reach goals and outcomes
3. **Processes** (sequences of activities) to complete or deliver services.
4. **Roles, data** and other structural resources needed to perform processes
5. **Actors** and organizations to perform roles and manage resources.

Active structures or building blocks

- TOGAF defines a building block as a “package of functionality” definable by its interface.
- This definition may be applied to
  - Components
  - Functions
  - Organization units
  - Roles
  - Human and computer actors

A Business Scenario that involves successive customer interactions may be called a **Customer Journey.**

Mapping human and computer actors to process steps



This extract from architect training at http://avancier.website
can be found at https://lnkd.in/gAGM_p2

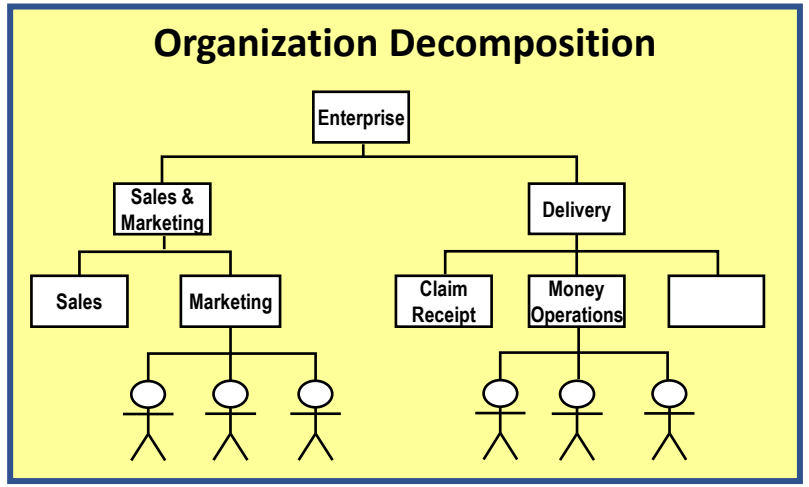
Simplified business architecture method

1. **Goals** or outcomes of interest to external actors and sponsors
2. **Services** (entry and exit conditions) external actors need to reach goals and outcomes
3. **Processes** (sequences of activities) to complete or deliver services.
4. **Roles, data** and other structural resources needed to perform processes
5. **Actors** and organizations to perform roles and manage resources.

Business architecture usually defined by managers

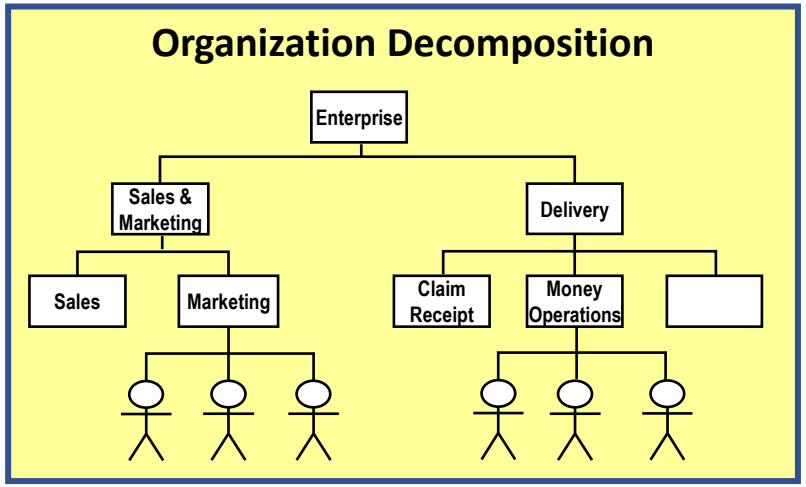
Organization (management) structure

- Typically decomposed from the top down
- May be supplemented by some “matrix management”
- May be stable or volatile, as a whole or in parts
- Typically, a unit has a manager, responsible for
  - Budget, sociological, psychological and HR matters
  - Monitoring and reporting of actors’ activities



How is an organization structure defined?

- Countless different structures can be imposed on the network of actors and activities in a large business or system.
- The "physical" management hierarchy clusters human actors into organization units according to a more or less clearly defined combination of criteria (location, goal, service type, skill and resource type, customer type etc.).
- Cohesion criteria differ at different levels.
- Directors manipulate the structure in the light of shifts in which cohesion criteria they think most effective, and other factors such as which managers they trust.
- EA methods presume a function hierarchy is more stable.



Mapping organization units to functions

"A business function represents a *collection* of business behavior based on chosen criteria.... closely aligned to an organization.." ArchiMate 3.1

"Closely" is misleading.

- A function is realized by one or more Organization Units.
- In a “functional organization” the relationship is 1 to 1

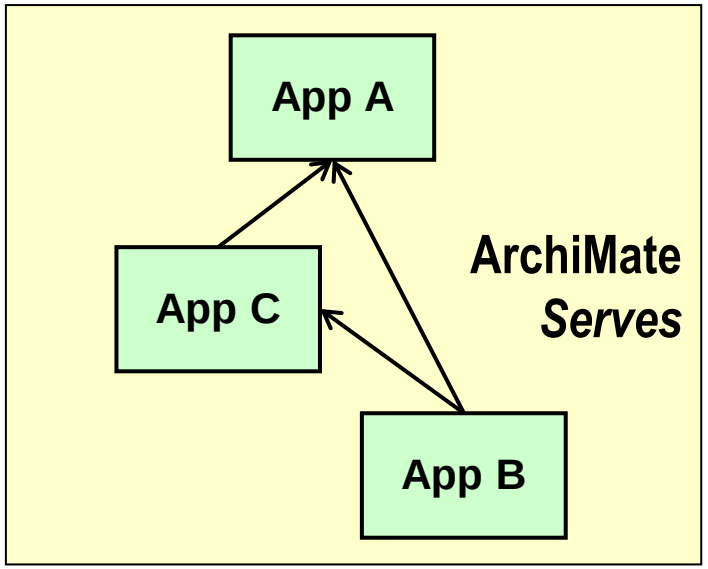
	Org	Org	Org
Function	x		
Function		x	
Function			x

- Else, the organization may be divided by region, resources, customer or product type, and mapped to functions in a matrix.

	Org	Org	Org
Function	x	x	
Function		x	x
Function	x		x

Inter-system dependencies – serves relationship

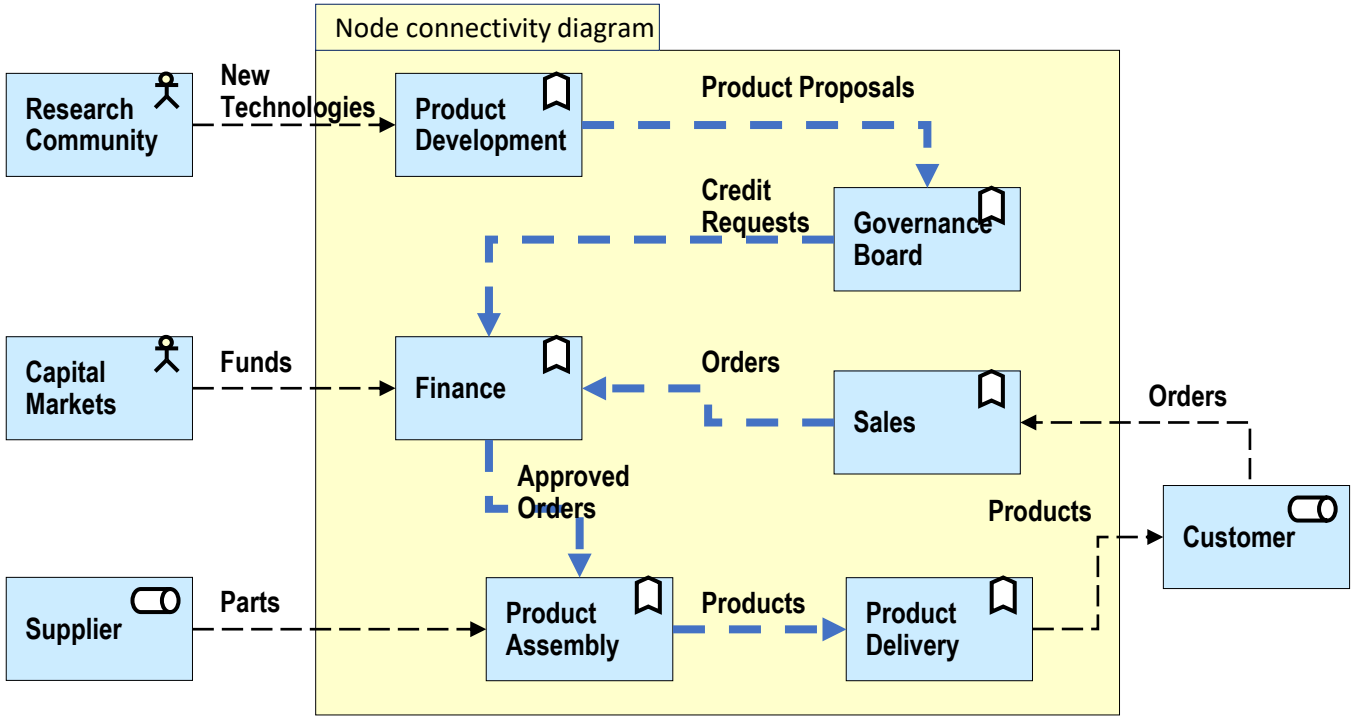
- A system may depend on other systems, with which it exchanges materials and/or information
- In ArchiMate, you can represent an inter-system dependency at an abstract level by a ***serves*** arrow between boxes



Inter-system dependencies - data flows

- High (2nd or 3rd) level functions or capabilities usually cannot be sequenced in a process
- Instead they can be related structural view **data flows (**rather than triggers).

- (Discrepancies between languages used in different subsystems may be resolved by standardization, or by transformers or adapters applied to data structures in flows between senders and receivers.)



Decomposing and relating architecture concepts

- Since systems can be nested, goals, services, functions, processes, roles, components, data structures, locations etc. can all be composed and decomposed into bigger and smaller entities of the same kind - from the very top of an enterprise down to atomic activities and actors not further described.

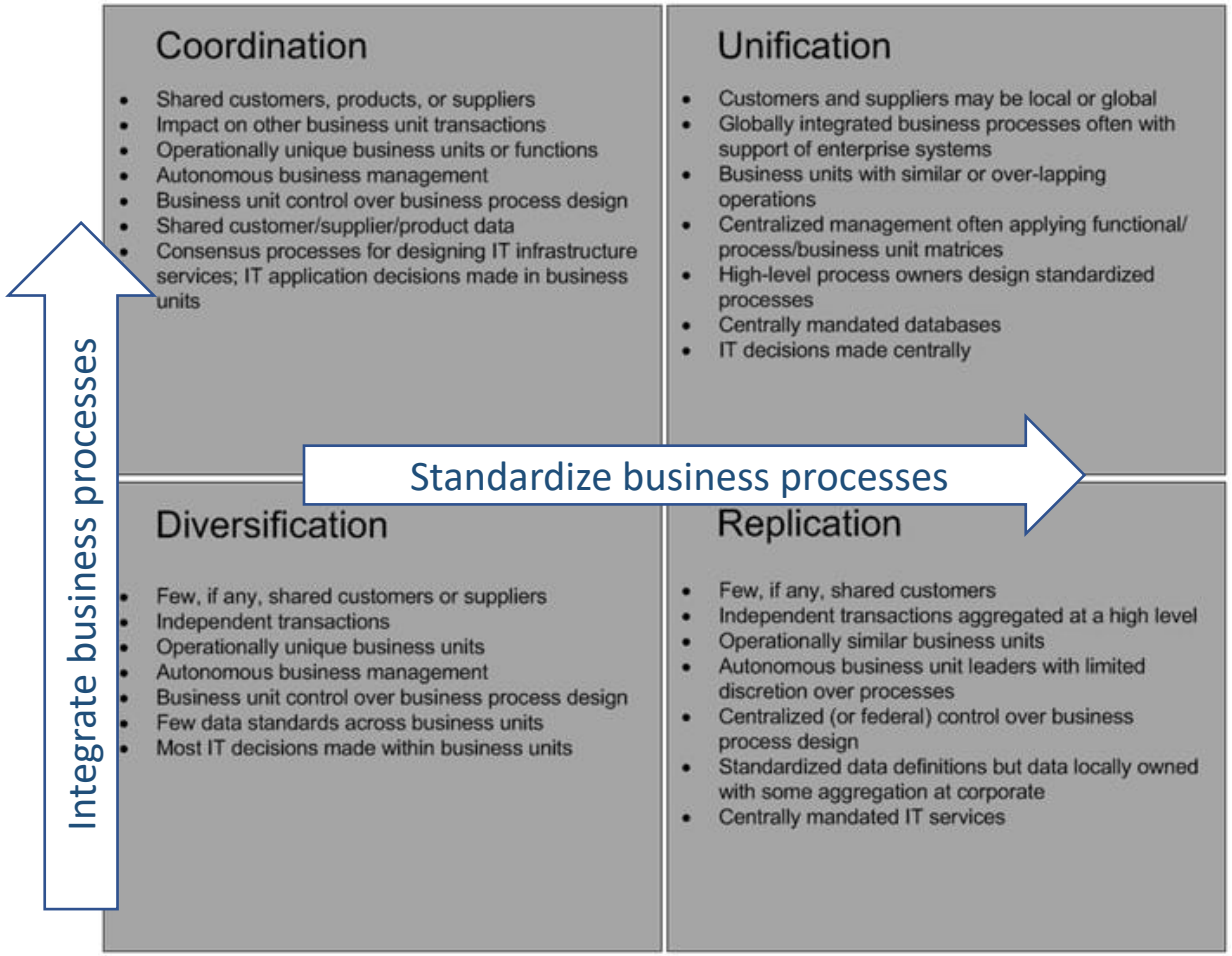
- Using different words for the same concept at different levels is a source of confusion.

- Given hierarchical decomposition (and other reasons) entities of different types are generally related N-to-N.
- E.g. One actor can play many roles; one role can be played by many actors.

Additional remarks

“Choose your Operating Model”

(Says “Enterprise Architecture as Strategy” Ross et al, 2006)



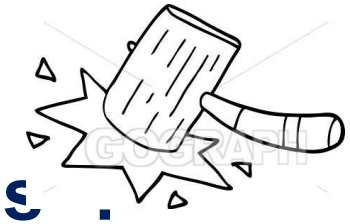
EAs look not only to improve the efficiency and effectiveness of business processes

- standardize business processes
  - which implies standardizing data
- integrate business processes
  - which implies sharing/exchanging data

but also to improve the creation and use of business data

- standardize and consolidate data
- improve data qualities (CIA)
- capitalize on data captured
- enable cross-organizational data analysis.

This extract from architect training at http://avancier.website
can be found at https://lnkd.in/gAGM_p2



“An enterprise is a system of systems”.

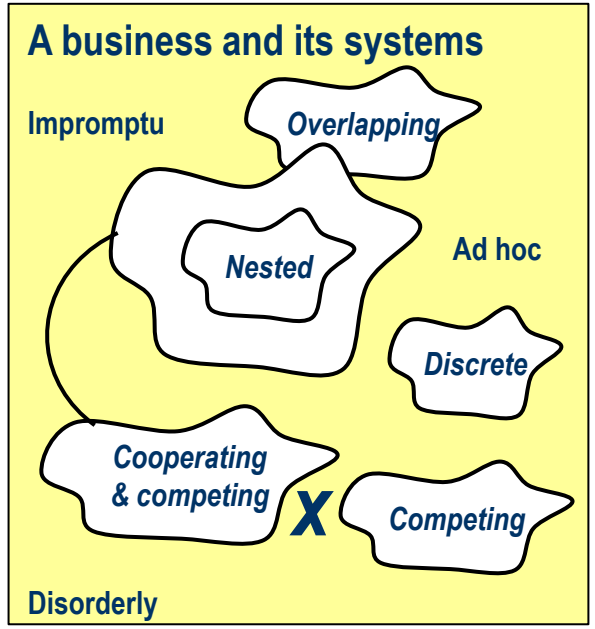
A business is

- an infinitely complex social network in which people often act in ad hoc ways
- unlike a homeostatic animal, it *progresses* state variables
- unlike a machine, it is only system-like in particular behaviors.

A business is **a mess of systems** in that it

- employs many systems, coordinates *some* of them
- progresses many state variables, regulates *some* of them
- extends, changes and digitizes its systems incrementally.

EA strives to tidy up the mess, but cannot standardize, coordinate and digitize *all* activities.



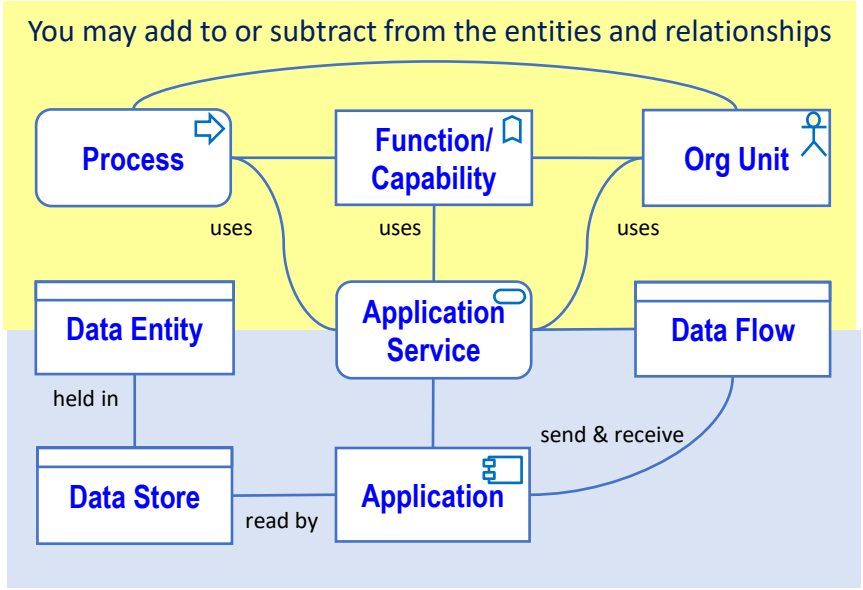
Given a mess of business systems – where to start?
Use EA concepts to help you understand the mess

- Identify your organization structure
- Identify your functions (or capabilities) and map them to organization units
- *Look for functions duplicated in different organization units*

- Identify cross-organizational processes and draw them in simple high level diagrams
- Identify applications and map them to functions and/or org units and/or processes
- *Look for duplicate applications*

- Identify data stores and map them to applications
- Identify core/kernel data entities in the data models of the data stores
- *Look for duplicate kernel data entities (e.g. employee, product) and common reference data (e.g. exchange rates)*

- Identify data flows and draw one or more application communication diagrams
- Identify cross-organizational **duplications, disintegrities, delays and difficulties with data analysis**
- Focus on business services and processes that are problematic or new



Finding duplicate applications

Where **one high level function or capability is mapped to two or more organization units**

- because the organization is structured using some other criteria (like region, customer or product type)

you may find duplication of applications across the enterprise.

	Org	Org	Org
Function	x	x	
Sales		x	x
Function	x		x

Where **decomposition to a lower level reveals common functions or capabilities**

- atomic activities (like Complete Timesheet or Write Letter)
- wider functions (like Work Scheduling, Ledger Maintenance)

you may find duplicate applications there also.

So, your function catalog may contain functions not in a FDD.

On communicating business architecture concepts

- You don’t have to model everything about a business that can be documented.
- You model what you need to understand and explain.
- Then choose what to communicate - simplifying as need be.

- These slides address terms and concepts EAs may use to
  - organize their thinking
  - organize EA data that is documented
  - discuss unambiguously with other trained architects.

- Talk to business leaders and others in *their* terms - translating as need be.

How to map TOGAF's business architecture value stream to its ADM?

This diagram maps activities and/or building blocks to process stages

And maps process stages to the phases (A to G) of the ADM

The first three stages might done as "free standing exercise" in the Preliminary Phase, before an ADM cycle.

The first two stages include things business managers may do.

Business management consulting?

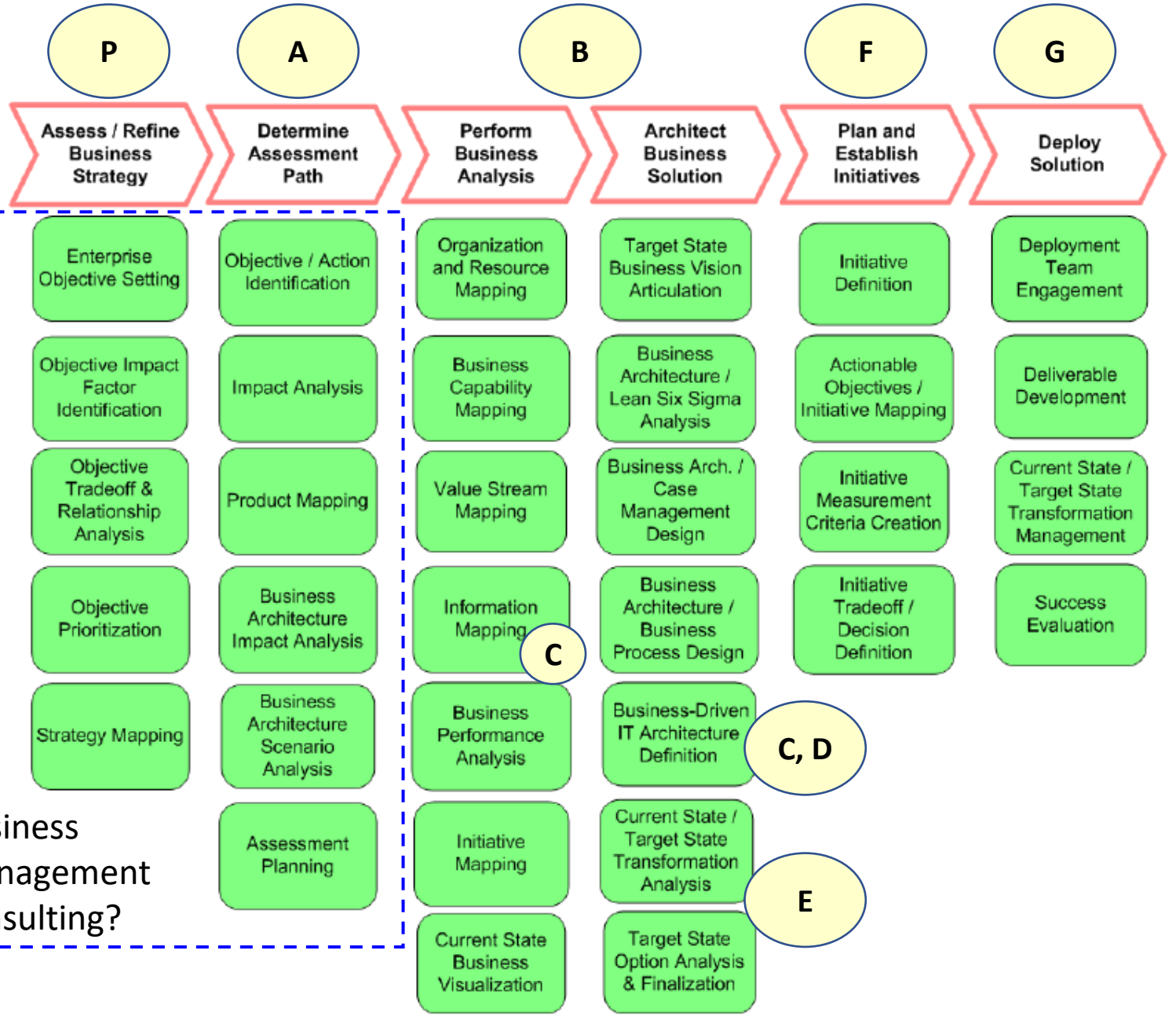


Figure 1.4: The Business Architecture Value Stream

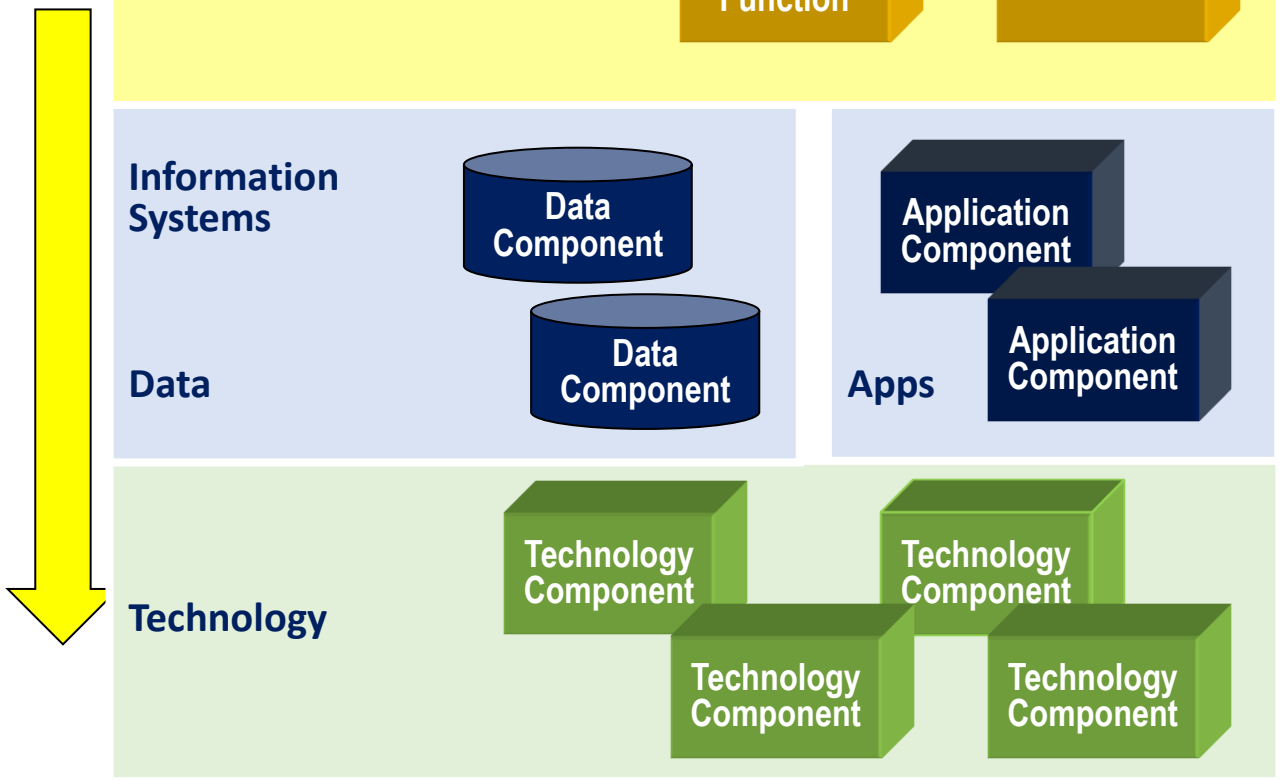
This extract fr
can

More on principles of architectural thinking

1. Business before technology
2. External before internal
3. Behavior before structure
4. Logical before physical

Principle 1: Business before technologies

- A business procures technologies to enable its business operations.
- The EA principle is to consider first what must be enabled.



Principle 2: External before internal

- The internal structures and behaviors of business are designed to produce the externally recognizable results, outputs of services that its customers, consumers or users require.

- Services encapsulate activities/processes. Interfaces encapsulate actors/components

	Behavioral view	Structural view
External view	Service contracts	Interface definitions
Internal view	Activities / Processes	Actors / Components

- Seeing an interface definition as a logical component is one way to harmonize TOGAF and ArchiMate.

Services

- A **service** a discrete behavior that leads to a result of benefit/value to an entity outside the system of interest.
- It can provide to an external entity.
  - goods (a pizza), information (a train ticket), a state change (shiny shoes),
  - or a mix of goods, information and state changes,
- E.g.
  - The "Replace tyre" service gives you a new tyre.
  - The "Polish my shoes" service gives you shiny shoes.
  - The "Book train ticket" service gives you a paper ticket and/or a digital ticket
  - The “Turn data into Insights” service gives managers insights into their business

- Cf. UML, "A use case… yields an observable result.. of value for … stakeholders....”

Principle 3: Behaviors before structures

- The structures of a business are built, hired or bought to perform or enable the behaviors required of that business.

	Behavioral view		Structural view
External view	Service contracts	→	Interface definitions
Internal view	Activities / Processes	→	Actors / Components

- A service may be accessed via an interface, or triggered by a state change. E.g.
  - You may request a delivery of a pizza via an interface.
  - Your butler polishes your shoes, triggered by the condition of your shoes.

Principle 4: Logical before physical

- A logical component specifies what a physical component should do
  - (the services it should provide, and perhaps the data it must maintain)
- Regardless of any internal actors or technology.

	Behavioral view	Structural view
External view	Service contracts	Interface definitions
Internal view	Processes	Logical Components Physical Components

Where does business architecture fit?

- Value Streams are Processes
- Functions and Capabilities can be seen as Logical Business Components
- Organizations and Actors can be seen as Physical Business Components

	Behavioral view	Structural view
External view	Service contracts	Interface definitions
Internal view	Value Streams	Functions, Capabilities, Roles Organizations, Actors

- Physical actors are hired to play logical roles.
- Physical organization units are managed to realize logical functions or capabilities.

Easing the terminology torture – after ArchiMate

This grid classifies concepts used in modelling business systems, using the categories in ArchiMate.

	Behaviors	Structures	
External view	A behavior defined by its entry and exit conditions, as external entities see it. Business Service	A declaration of available and accessible services. Business Interface (SLA)	
Internal view	A behavior defined as a flow of stages or steps from start to end. Process/ Value Stream	A logical division of business behavior, grouping related activities Function/ Capability Role	**Logical**
		A physical structure capable of performing behaviors Organization Unit Actor	**Physical**

A classification compatible with ArchiMate and TOGAF

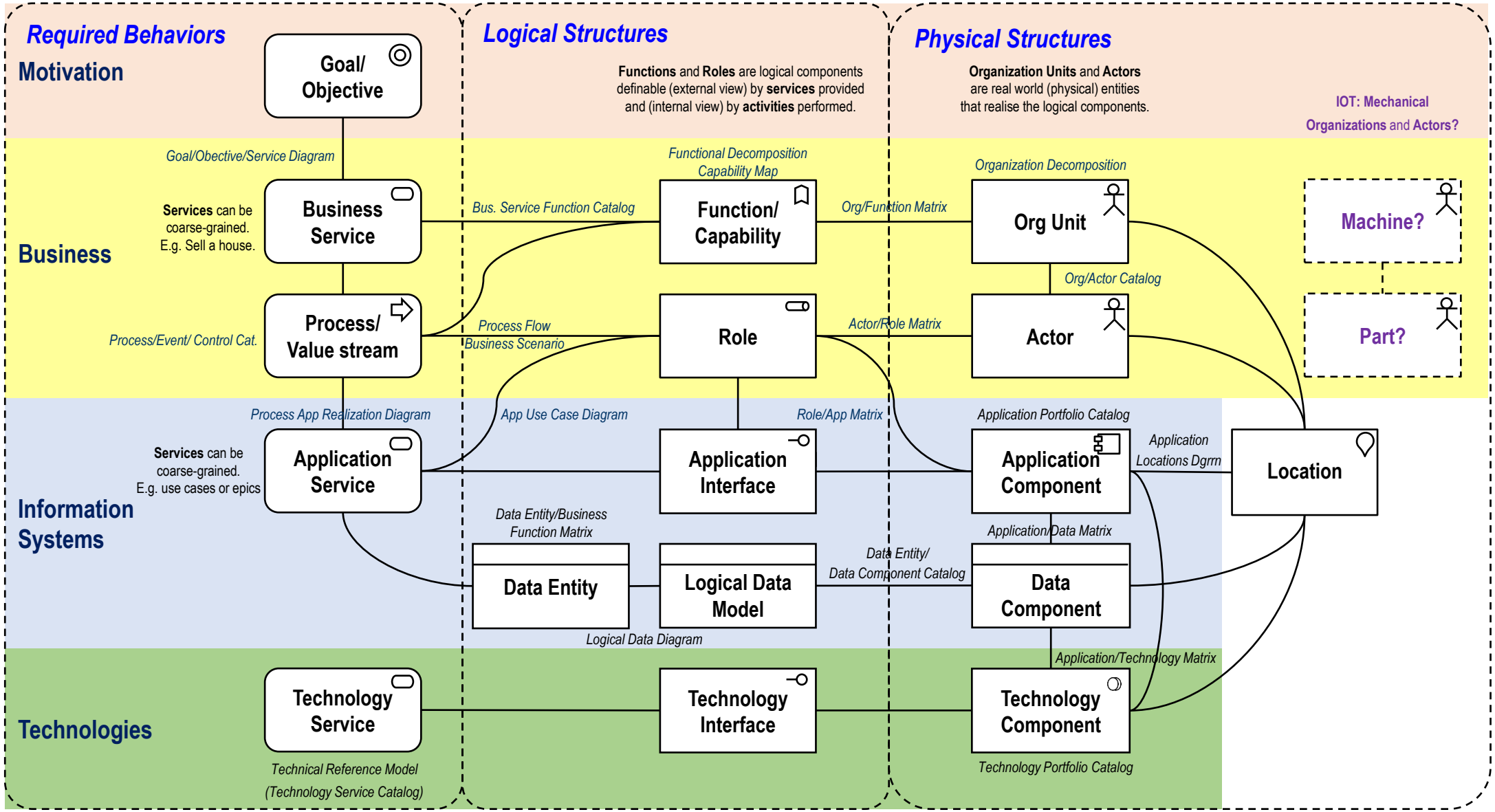
Four principles embodied in ArchiMate and TOGAF

1. Business before technology
2. External before internal
3. Behavior before structure
4. Logical before physical

	Behaviors	Structures	
Business architecture Services that a Business provides to its Customers (to meet Goals/Objectives) are important to most BA and EA practice.	Business Service Process/ Value Stream	Business Interface Function/ Capability Role Organization Unit Actor	*Logical* *Physical*
Applications architecture Services (use cases, epics) that Applications provide should be identified by Solution Architects, but may be obscure in more abstract EA practice.	Application Service	Application Interface Application Component	*Logical* *Physical*
Technology architecture Services that Technologies provide (as catalogued in TOGAFs TRM) are obscure in most modern EA practice	Technology Service	Technology Interface Technology Component	*Logical* *Physical*

This extract from architect training at http://avancier.website
can be found at https://lnkd.in/gAGM_p2

A meta model that makes sense of TOGAF 9



This extract from architect training at http://avancier.website
can be found at https://lnkd.in/gAGM_p2

Capabilities?

The need for a logical organisation structure

- EA is supposed to be cross-organizational
- The organization's physical management structure is often volatile/fluid
- It may be restructured (say by region, customer type or product type).

- Whereas the nature of the business (services, processes and data) is more stable.
- So, EAs often start by building a logical organisation structure, called
  - A Functional Decomposition Diagram, or
  - A Capability Map

Functional decomposition diagram (TOGAF)

"Shows on a single page the capabilities of an organization relevant to the consideration of an architecture.

By examining the capabilities of an organization from a functional perspective, it is possible to quickly develop models of what the organization does without being dragged into extended debate on how the organization does it.

Once a basic diagram has been developed, it becomes possible to layer heat maps on top of this diagram to show scope and decisions (e.g. the capabilities to be implemented in different phases of a change program.)" TOGAF 9.2

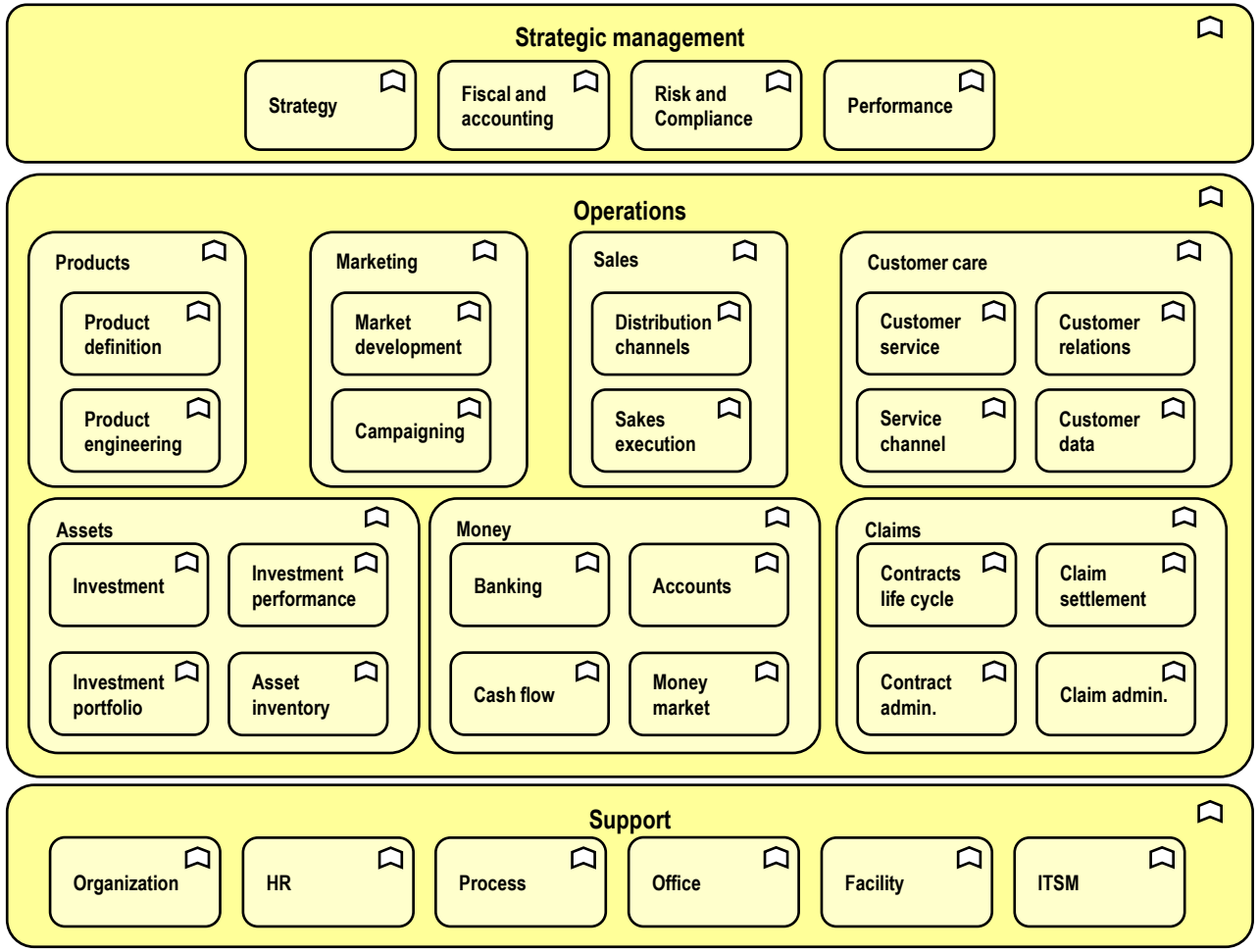


Diagram adapted from an ArchiMate example

Capabilities?

- A capability is the ability to
  - **do something** (implies activities are known) or
  - **achieve something** (implies aims are known)

- This video https://lnkd.in/eEsFdfR equates capabilities to subsystems of a human activity system, where a subsystem is defined as some actors using some resources to
  - **perform a known activity** to **achieve a known aim**

- The video is fine as far as it goes, but omits to mention things gurus don't discuss.

How do capabilities relate to functions?

- A capability usually corresponds to one or more of
  - A goal
  - An outcome
  - A function

- A function *groups activities* that are logically cohesive in some way.
- A capability *groups the abilities or resources* needed to perform activities that are cohesive in some way.

Business capabilities

"A capability represents an *ability* that an active structure element, such as an organization, person, or system, possesses." ArchiMate 3.1

It is hard it make practical use of that definition, but the standard goes on…

"*Capabilities* are expressed in general and high-level terms and are typically realized by a combination of *organization, people, processes, information, and technology*. For example, marketing, customer contact, or outbound telemarketing.“

The named examples could as well be the names of functions.

The elephant in the room?

- Simply put, to perform any function we name, we need the corresponding capability.
- In building a logical hierarchy, unless we choose different cohesion criteria for functions and capabilities (and why should we?), every capability will correspond to a function.
- Function and capability are not synonyms, but to build two hierarchies of elements that are in 1-to-1 correspondence is futile

This slide show has

Explored how four business architecture concepts

- Services
- Processes
- Functions
- Capabilities

relate to four principles of architectural thinking

1. Business before technology
2. External before internal
3. Behavior before structure
4. Logical before physical

Further reading

After decades research into these things, this slide show presents a distillation of the concepts behind the terminology torture and illustrate concepts in these two articles

- Service-Oriented Business Architecture
- Capability-Oriented Business Architecture

Other research includes

On business architect roles in the job market and in industry standards.

- “The practice of business architecture” https://bit.ly/2DRbCC0

On the science about what can be documented

- “The science of business architecture” https://lnkd.in/gBugTAX

On making sense of TOGAF’s concepts

https://lnkd.in/gTAh9iW

On systems

The term "system" is widely but loosely used in "systems thinking" and "management science", but best put those uses out of mind here, because the modelling techniques used in EA and BA are based on *system theory proper* as discussed here https://bit.ly/2w5XKNK .