This slide show should paginate nicely when viewed using Internet Explorer.
The show may be refined, so empty your browser cache before re-loading.

Avancier

TOGAF-ArchiMate alignment

Aligning **ArchiMate®** with **TOGAF®**

Warning: This slide show bends them both *a little*

Last updated 05/01/2017

A short preface

- Mainstream EA applies general system theory
- Mainstream EA takes a particular view of a business system
- Architectures are abstract descriptions of interacting components

Mainstream EA applies general system theory

Avancier

- “The principal heuristic innovation of the systems approach is what may be called ‘reduction to dynamics’ as contrasted with ‘reduction to components’ ” Laszlo and Krippner.

- EA defines the dynamics of a business in terms of *services* provided by *value streams/scenarios/processes/activities.*

- "Cybernetics does not ask "what is this thing?" but "what does it do?" It is thus essentially functional and behaviouristic.” Ross Ashby.

- EA presumes the requirement is for actors to perform roles that provide *services* required by system sponsors/stakeholders.

Architectures are abstract descriptions/models of interacting actors

- *"The method proposed by systems theory is to model… multiple interacting components by abstracting from certain details of structure and component."*

  (Laszlo and Krippner)

- *"Architecture' has two meanings depending upon the context:*
  - *1. A formal description of a system...*
  - *2. The structure of components, their inter-relationships..."* (TOGAF 9.1)

- The concept of interacting (services-providing) functions, roles or component types is central to enterprise architecture definition.
- This and a few other general concepts are fundamental to understanding TOGAF and aligning it with ArchiMate.

Contents

- Part 1: TOGAF's generic meta model
- Part 2: ArchiMate's generic meta model
- Part 3: Mapping the two meta models
- Part 4: TOGAF's entities and artifacts
- Part 5: More about abstraction

Part 1: TOGAF's generic meta model

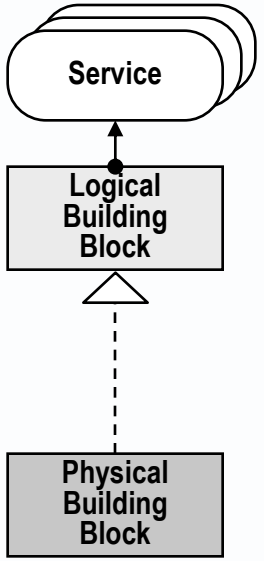
Aligning **ArchiMate®** with **TOGAF®**

Warning: This slide show bends them both a little

TOGAF's core principles

- An **enterprise** is a system of systems, that is:
  - a system of interoperating (services-providing) building blocks
  - with flows between them

- An **enterprise architecture** is an abstract (conceptual, logical) description/model of:
  - interoperating building block types
  - flow types between them



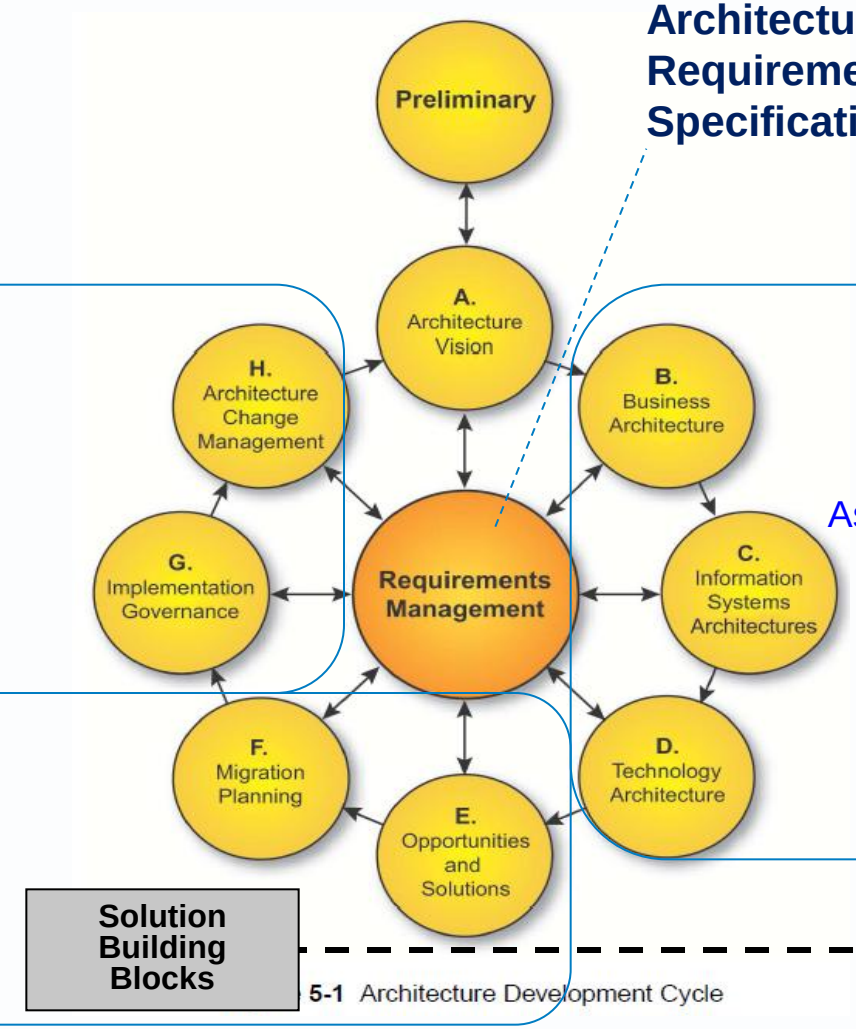
- Solutions architects get more physical, but still stop short of detailed design.

Abstraction of business system specification in TOGAF's ADM

Avancier

Governance of
solution delivery and change

Physical **Solution Options**
"physical elements…
may still be *considerably abstracted from Solution Architecture*"



Architecture Requirements Specification

Services

Logical **Architecture Definition Doc**

Assigned to provide

Architecture Building Blocks

Realised by

Realisation of logical elements by physical elements

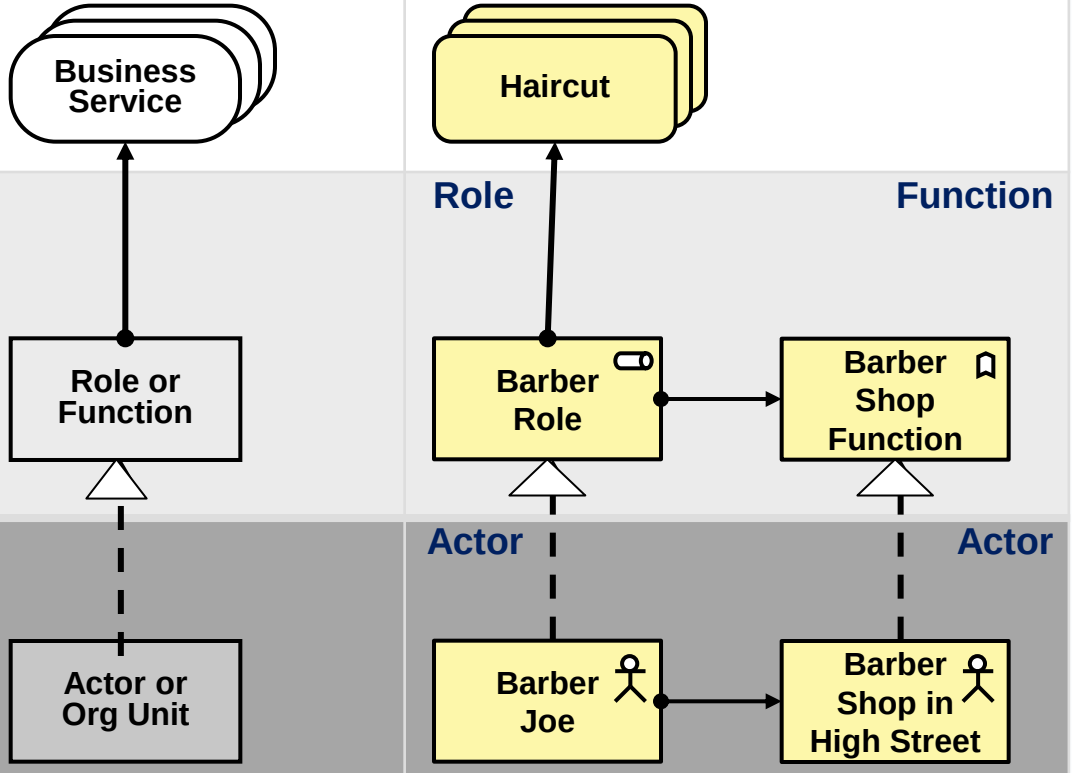
Avancier

Vocabulary	Required behaviours	assigned to	Logical structures	realised by	Physical structures
	Run over time		**Group behaviours**		**Do work**
TOGAF & ArchiMate	Business Services & Processes	←	Functions	◁	Organisation Units
		←	Roles	◁	Actors
Other	Value Streams	←	Capabilities		
TOGAF	IS Services	←	Logical Application Components	◁	Physical Application Components
ArchiMate	Application Services	←	Application Interfaces (or Functions)	◁	Application Components
Other	Use Cases & User Stories	←	User Interfaces	◁	Applications
TOGAF	Platform Services	←	Logical Technology Components	◁	Physical Technology Components
ArchiMate	Technology Services	←	Technology Interfaces (or Functions)	◁	Nodes (System Software & Devices)
Other			APIs		

TOGAF's Business Architecture

Avancier

TOGAF generic meta model	ArchiMate elements
Required behaviors (service portfolios) What services do customers want?	**Business service**
Logical structures What roles and functions are needed to provide the services?	**Role** **Function**
Physical structures What actors and organisation units can be acquired to play the roles and perform the functions?	**Actor** **Actor**



TOGAF's Data (aka Information) Architecture

Avancier

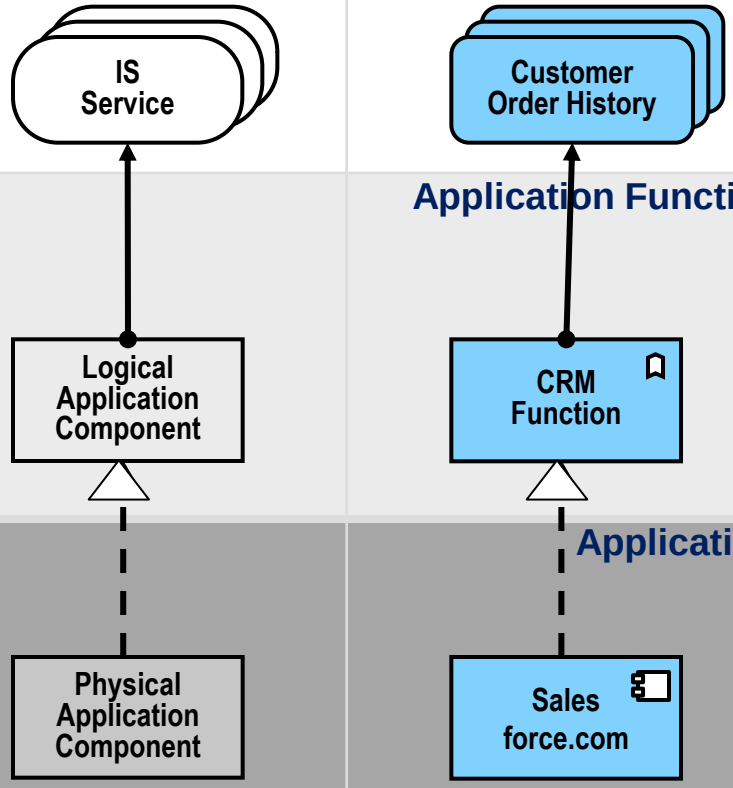
Data flows in service contracts and in what is called "Interface catalog"
Data models as below

Level of idealisation	Artifacts	
Conceptual data entities	Business/Conceptual Data Model	Things the business needs to remember (often duplicated in different systems).
Logical data entities	Logical Data Model	Data structure for discrete information system
Physical E.g. Tables	Physical data schema (e.g. Relational or Column Store)	Data structure for storage using selected technology
Real E.g. Rows in tables	Database	

TOGAF's Applications Architecture

Avancier

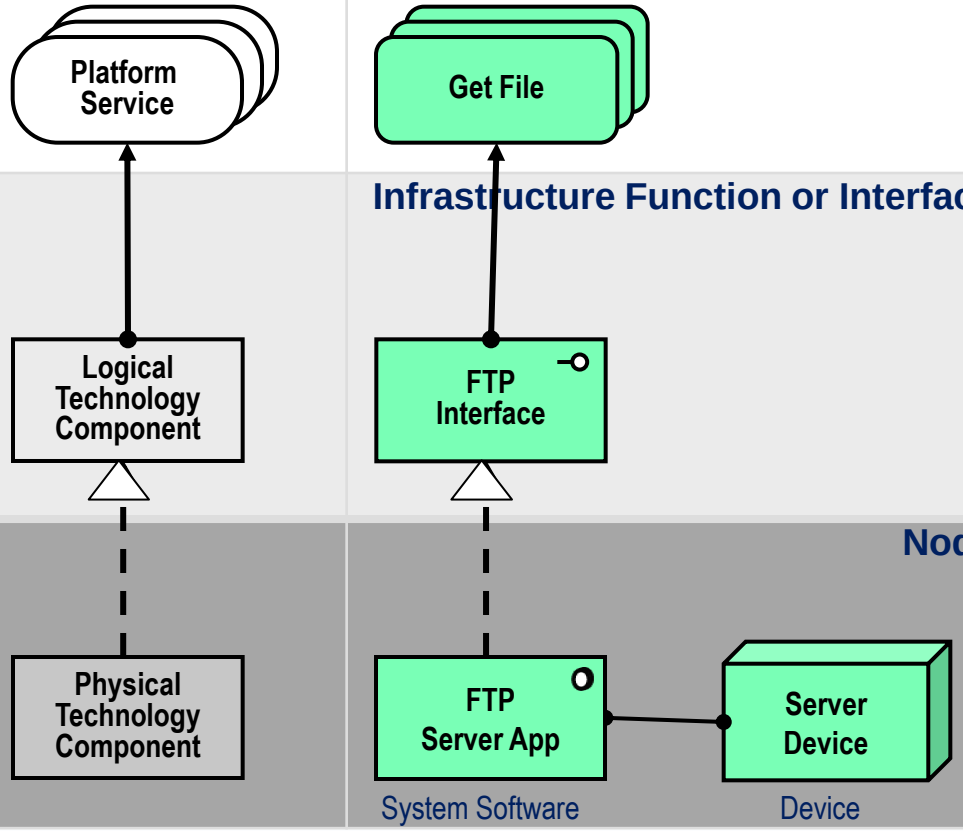
TOGAF generic meta model	ArchiMate elements
Required behaviors Your sales Organisation unit require twenty Services (or use cases) of an Information System	**Application Service**
Logical (vendor-neutral) **structures** You call it Customer Relationship Management. It could be realised by people, or by any of several COTS ("packaged") applications.	**Application Function or Interface**
Physical structures You choose one application because it offers 18 of the 20 of the required services. It offers 5 other services you never thought to ask for, which are "opportunities".	**Application Component**



TOGAF's Technology (Infrastructure) Architecture

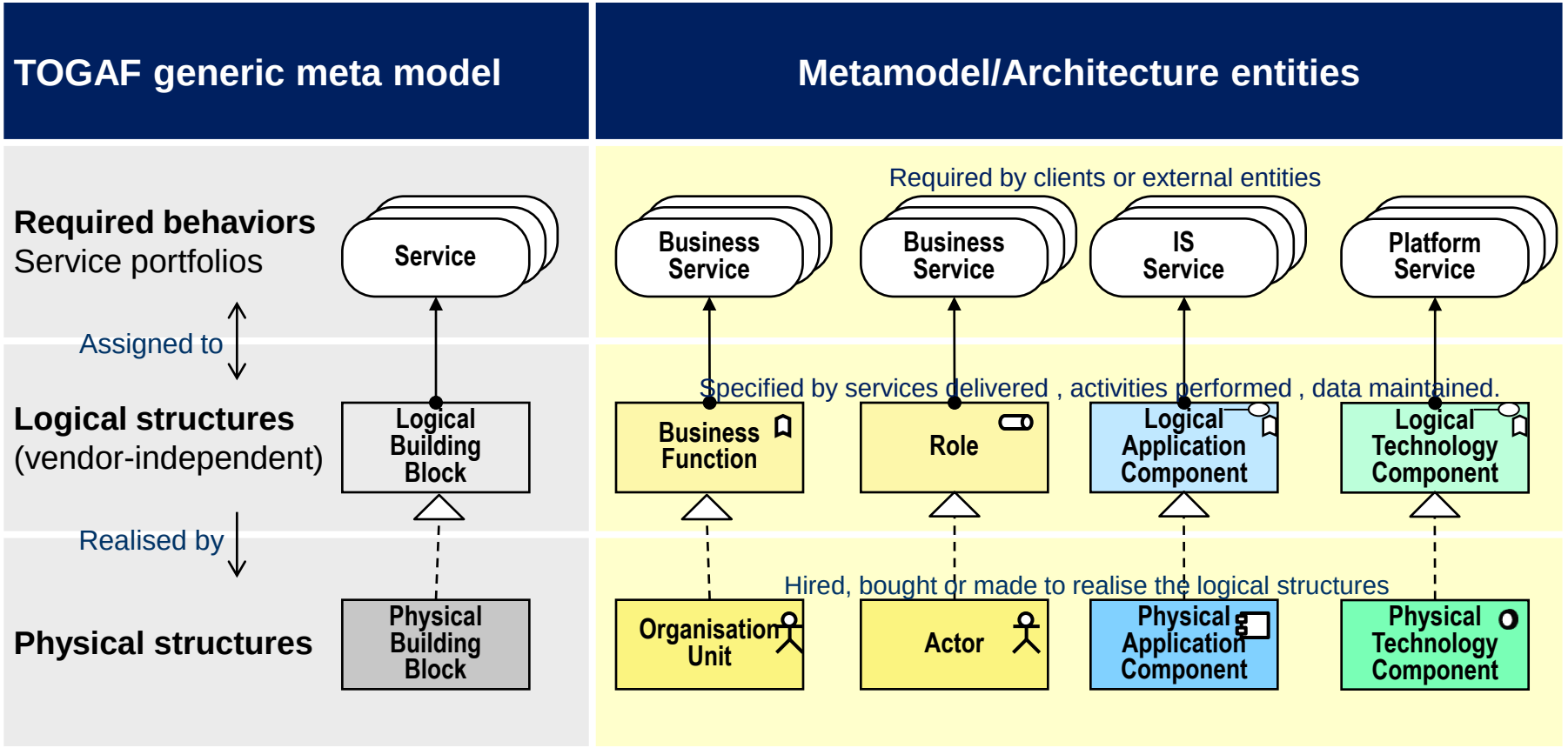
Avancier

TOGAF generic meta model	ArchiMate elements
Required behaviors Service portfolios Selected from the enterprise's TRM	**Infrastructure Service**
Logical (vendor-neutral) **structures** ABBs defined by "service portfolio" each provides E.g. the IETF standard FTP interface.	**Infrastructure Function or Interface**
Physical structures SBBs hired, bought or built to realise the ABBs. E.g. the particular FTP server deployed	**Node**



TOGAF's generic meta model using ArchiMate symbols

Avancier



Logical components may be modelled as functions or interfaces

A reasonable, though imperfect, alignment

- TOGAF's generic meta model

- ArchiMate's generic meta model

	Behaviors	Active Structures
External view	Service	Function or Interface
Internal view	Process	Building Block

"Typically, EA does not drill into process flow, but in certain process-centric or event-centric organizations it may be necessary" TOGAF

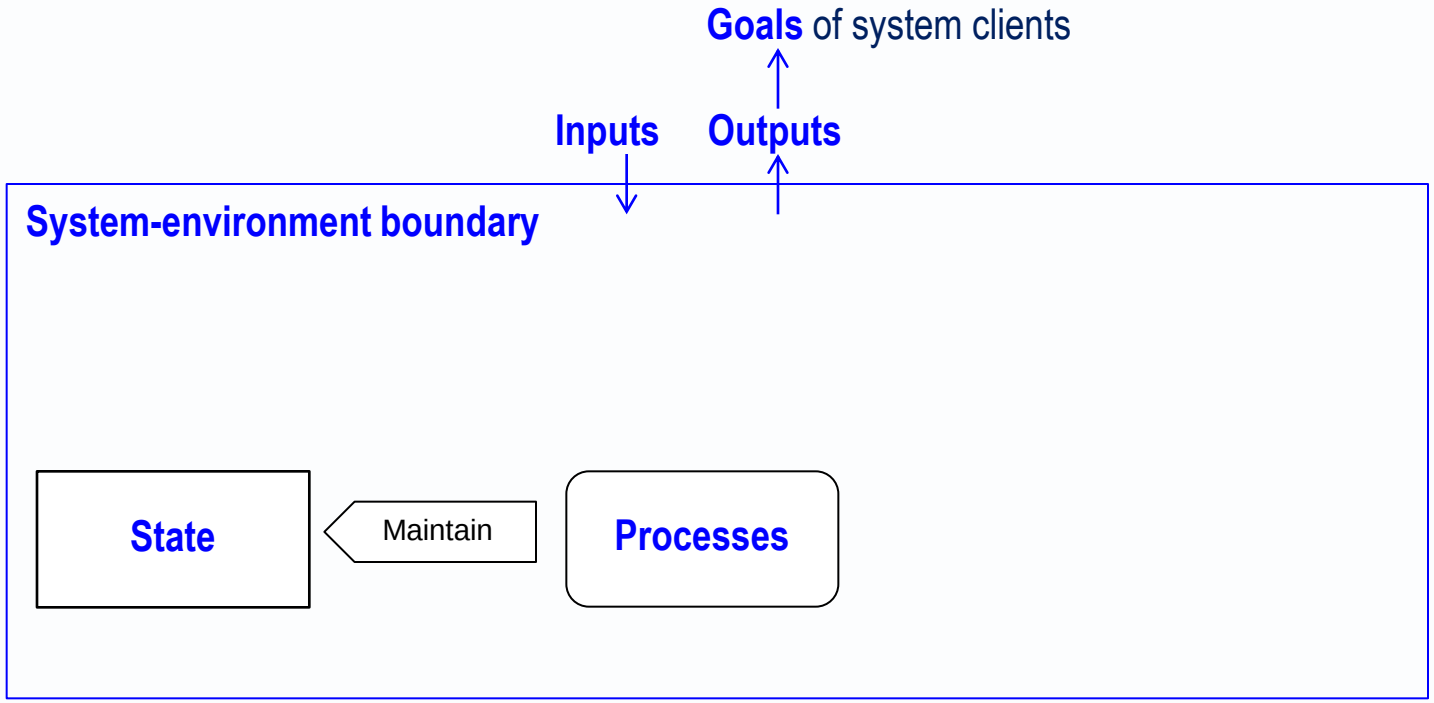
Part 2: Aligning generic meta models

Aligning **ArchiMate®** with **TOGAF®**

Warning: This slide show bends them both a little

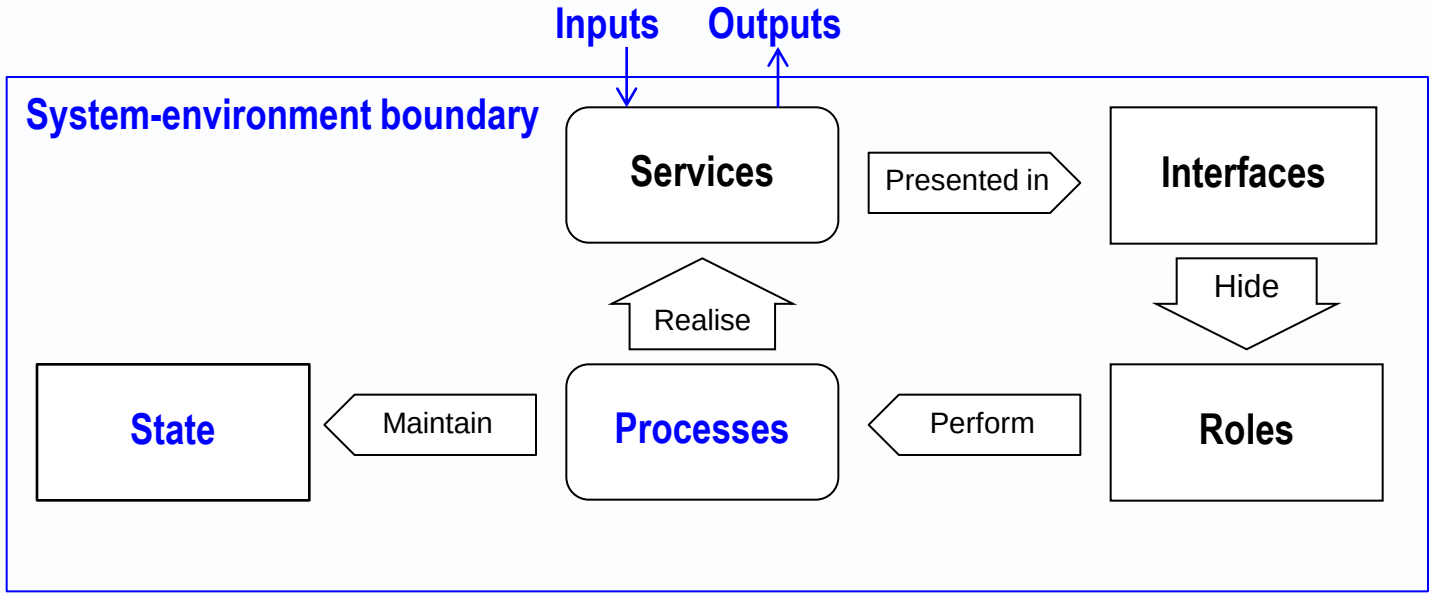
General system theory – a few core concepts

- "Systems concepts include: system-environment boundary, input, output, process, state… goal-directedness" Principia Cybernetica



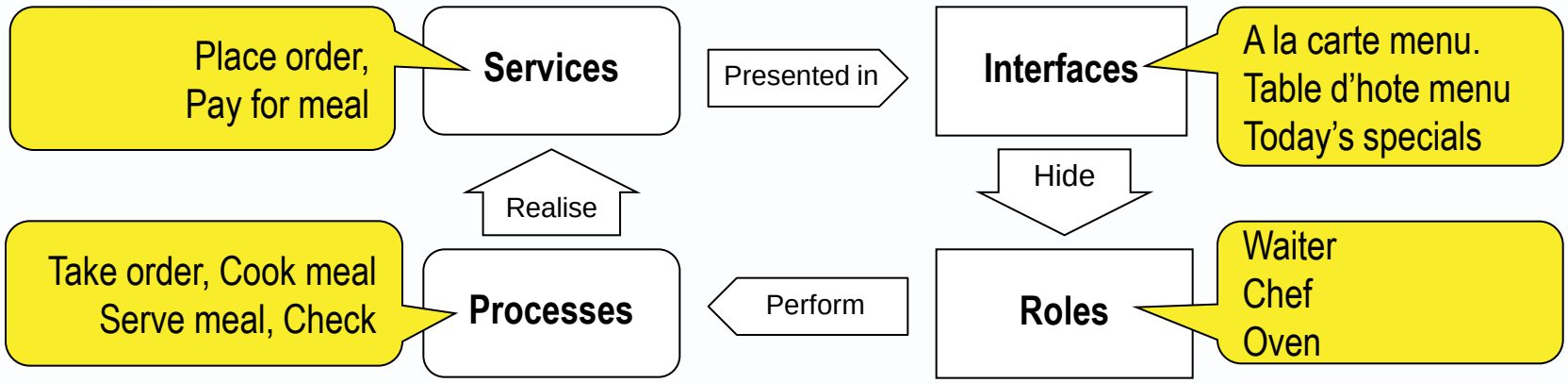
A business system theory – core concepts

- In a business you find actors playing **roles** in the performance of **processes** to provide regularly requested **services** which are presented to clients in **interfaces**



The business system theory applied to a restaurant

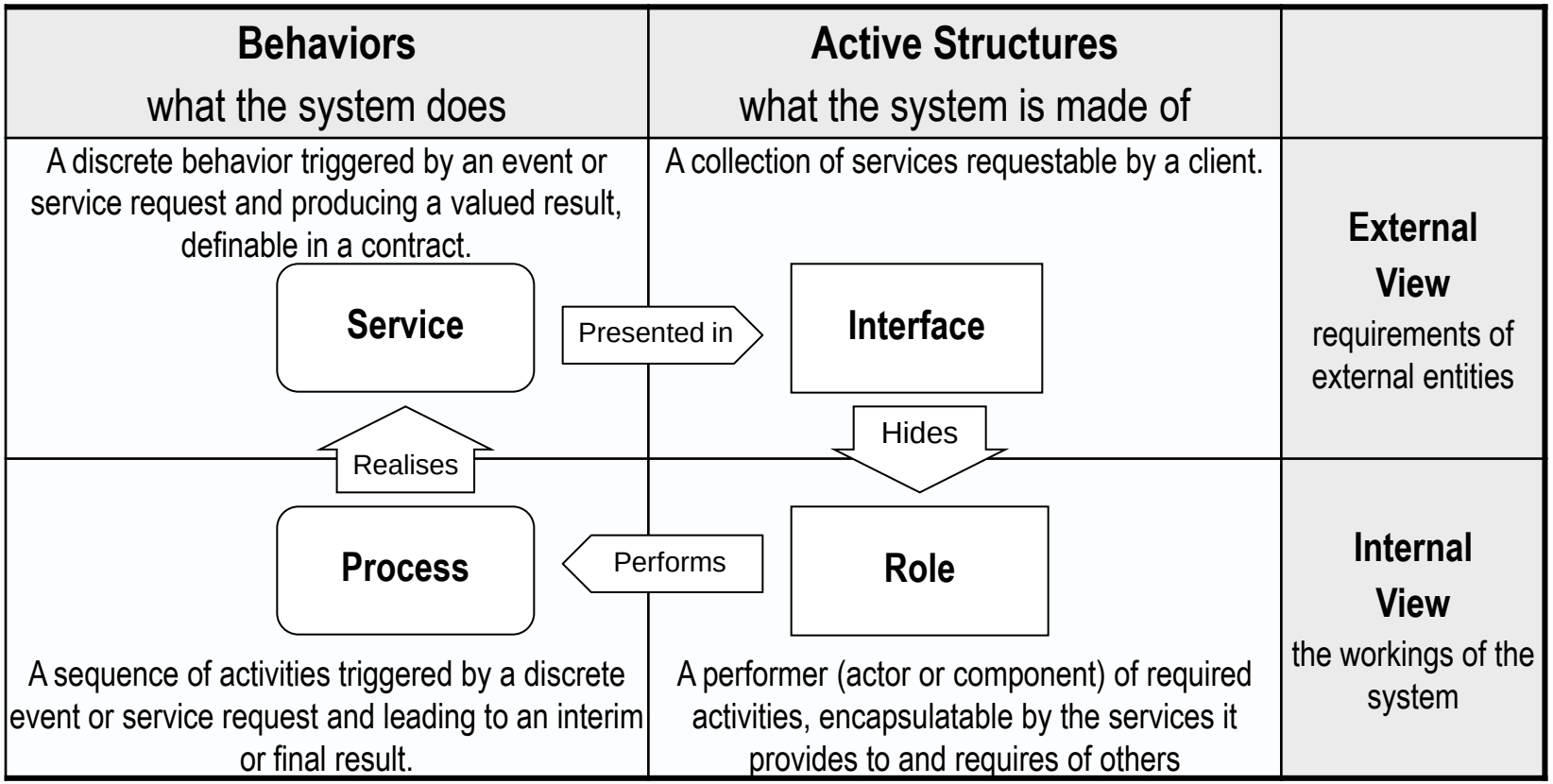
- The clients requirements are for **services**
- Rather than the internal **roles** and **processes** needed to provide them



Concepts used in modelling business systems

Avancier

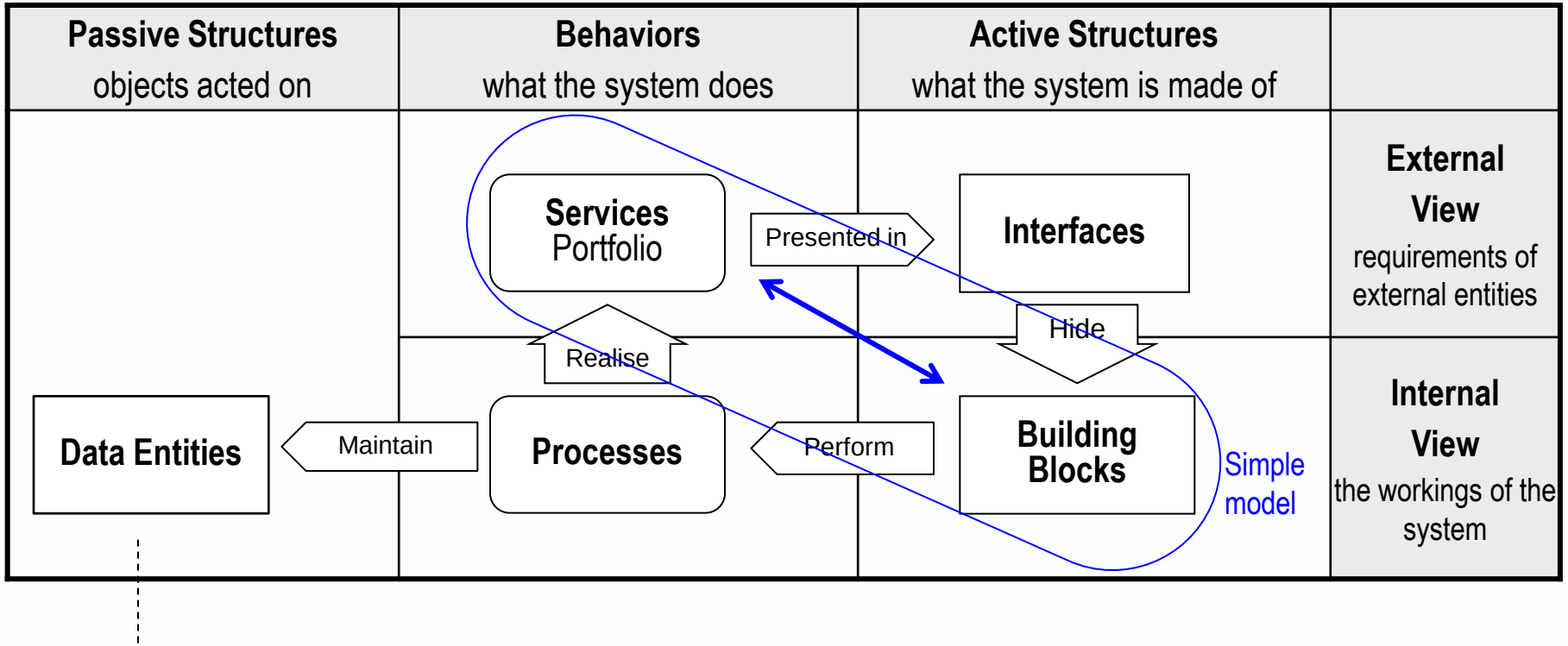
- Structure is being; behavior is doing



All associations here are many-to-many

The basis of TOGAF: services-providing building blocks

- "Systems are built up from … **building blocks** [that] interoperate with other building blocks"
- "For each building block, build up a **service description portfolio**"
- "It is important that the **interfaces** to a building block are published and reasonably stable"

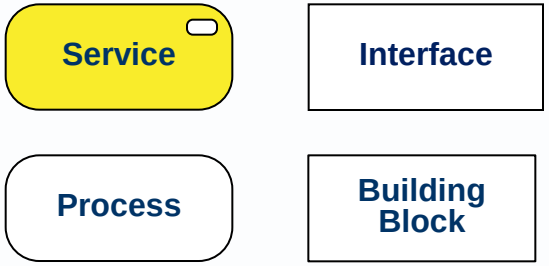


- Business state data is created and used by business processes

“Service” in TOGAF and ArchiMate

Avancier

A discrete behavior triggered by an event or service request and producing a valued result; definable in a contract

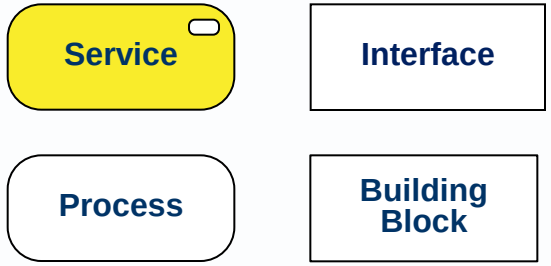


- TOGAF: “Check customer credit: Provide weather report, Consolidate drilling reports.”
- ArchiMate: “Policy Creation, Claim Registration, Claim Payment.”

TOGAF says	ArchiMate says	Generally speaking
“an element of behaviour that provides specific functionality	“a unit of functionality that	A discretely requestable behaviour (short or long) that
in response to requests from actors or other services.	a system exposes to its environment,	is triggered by an event or service request
A logical representation of a repeatable business activity,	hides internal operations,	is definable in a contract that encapsulates processing,
has a specified outcome,	provides a value,	produces a valued result and
is self-contained, is a ‘‘black box’’ to its consumers.”	accessible through interfaces.”	is presented for access in one or more interfaces.

A service contract template

- ArchiMate
  - “exposed functionality and value,
  - together with non-functional aspects
    - such as quality of service, costs…
  - These can be specified in a contract.”

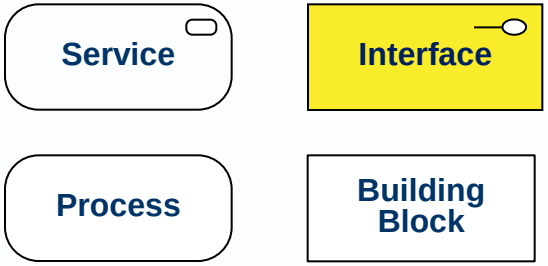


- A service contract template:
  - Signature
    - Name, input flow, output flow (which provides value)
  - Functional rules
    - Preconditions
    - Post conditions (which also provide value).
  - Non-functional characteristics
    - Performance, Availability, Security etc.
    - Commercial conditions.

“Interface” in TOGAF and ArchiMate

Avancier

A collection of services requestable by a client

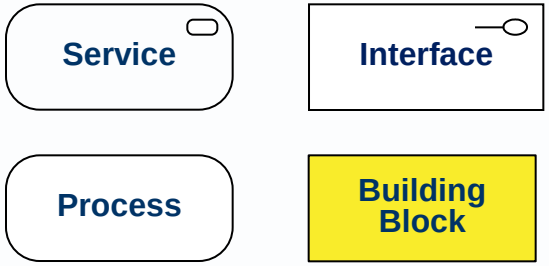


- TOGAF
  - tends to use “interface”, “boundary” and “service portfolio” interchangeably.
- ArchiMate
  - “a point of access where services are made available”.

“Building Block” in TOGAF
cf. “Internal Active Structure Element” in ArchiMate

Avancier

A performer (actor or component) of required activities, encapsulatable by the services it provides to and requires of other building blocks.



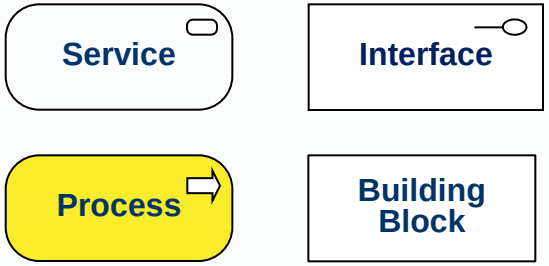
TOGAF 9 chapter 37 says:
Systems are built up from … building blocks [that] interoperate with other building blocks.
For each building block, build up a service description portfolio as a set of non-conflicting services.
It is important that the interfaces to a building block are published and reasonably stable.

- a building block is generally recognizable as "a thing" by domain experts
- is a package of functionality defined to meet the business needs across an organization.
- has published interfaces to access the functionality.
- may interoperate with other, inter-dependent, building blocks.
- considers implementation and usage, and evolves to exploit technology and standards.
- may be assembled from other building blocks.
- may be a subassembly of other building blocks.
- is ideally re-usable and replaceable, and well specified.
- may have multiple implementations but with different inter-dependent building blocks.
- is therefore simply a package of functionality defined to meet business needs.

“Process” in TOGAF
cf. **“Internal Behavior Element” in ArchiMate**

Avancier

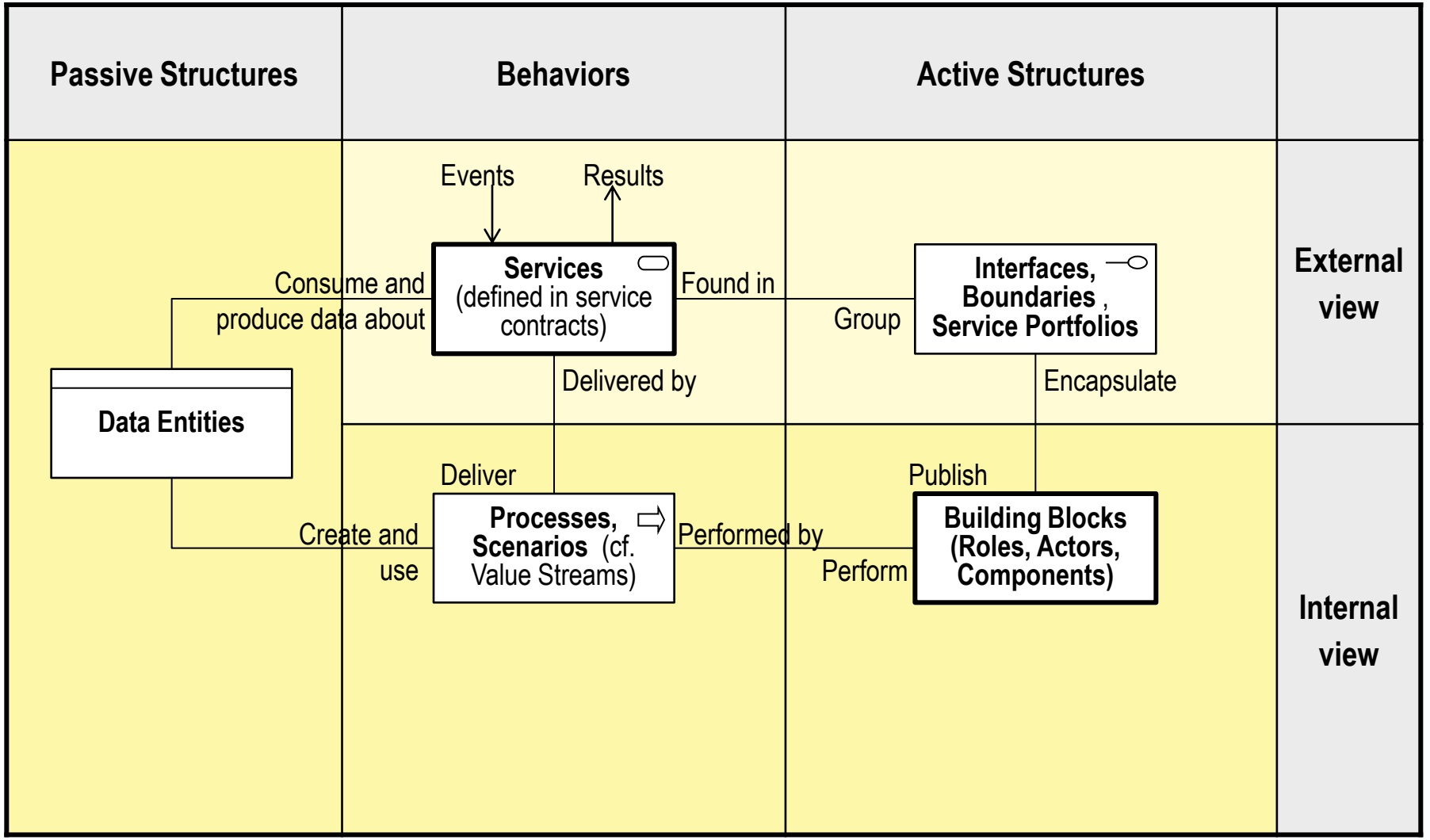
A sequence of activities, triggered by a discrete event or service request, that lead to an interim or final result.



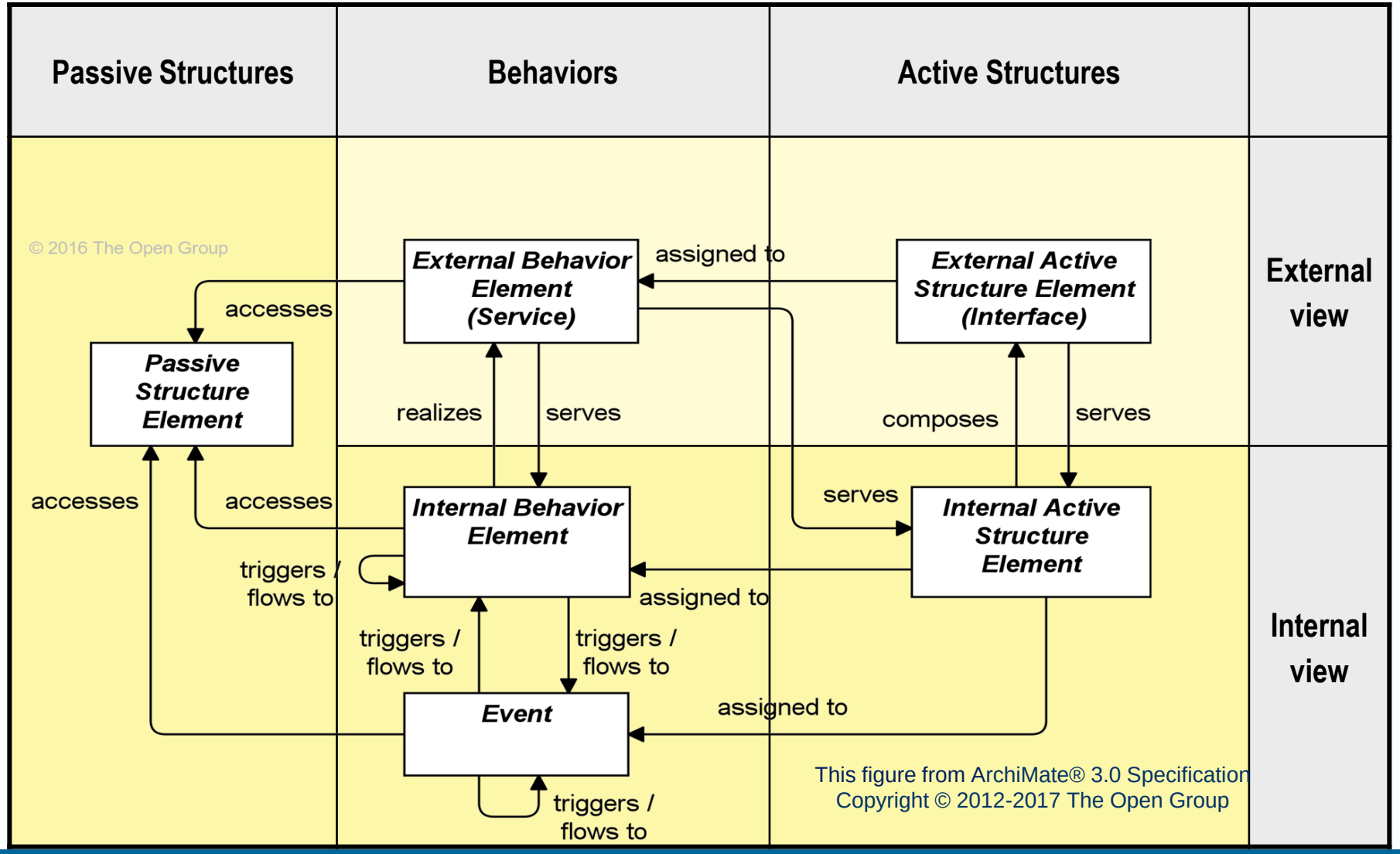
- TOGAF
  - “A flow of interactions between functions and services”
  - “a sequence of activities that together achieve a specified outcome”
  - “can be decomposed into sub-processes
  - “Typically, EA does not drill into process flow, but in certain process-centric or event-centric organizations it may be necessary to elaborate process in a much more for mal manner”.

- ArchiMate
  - “a sequence of business behaviors that achieves a specific outcome such as a defined set of products or business services.”

Showing TOGAF concepts in this generic structure



Showing ArchiMate concepts in this generic structure



One more thing: Function

A subdivision an organisation's capability; a logical building block that groups behaviors by some criteria other than sequential flow.

- TOGAF
  - "Function describes units of business capability at all levels of granularity"
  - "Describes a unit of business capability at all levels"
  - "Any bounded unit of business function."
- ArchiMate
  - "a collection of business behavior based on criteria (typically required business resources and/or competences)"
  - "aligned to an organization, but not necessarily explicitly governed by [it]."

Part 3: Mapping the two meta models

Aligning **ArchiMate®** with **TOGAF®**

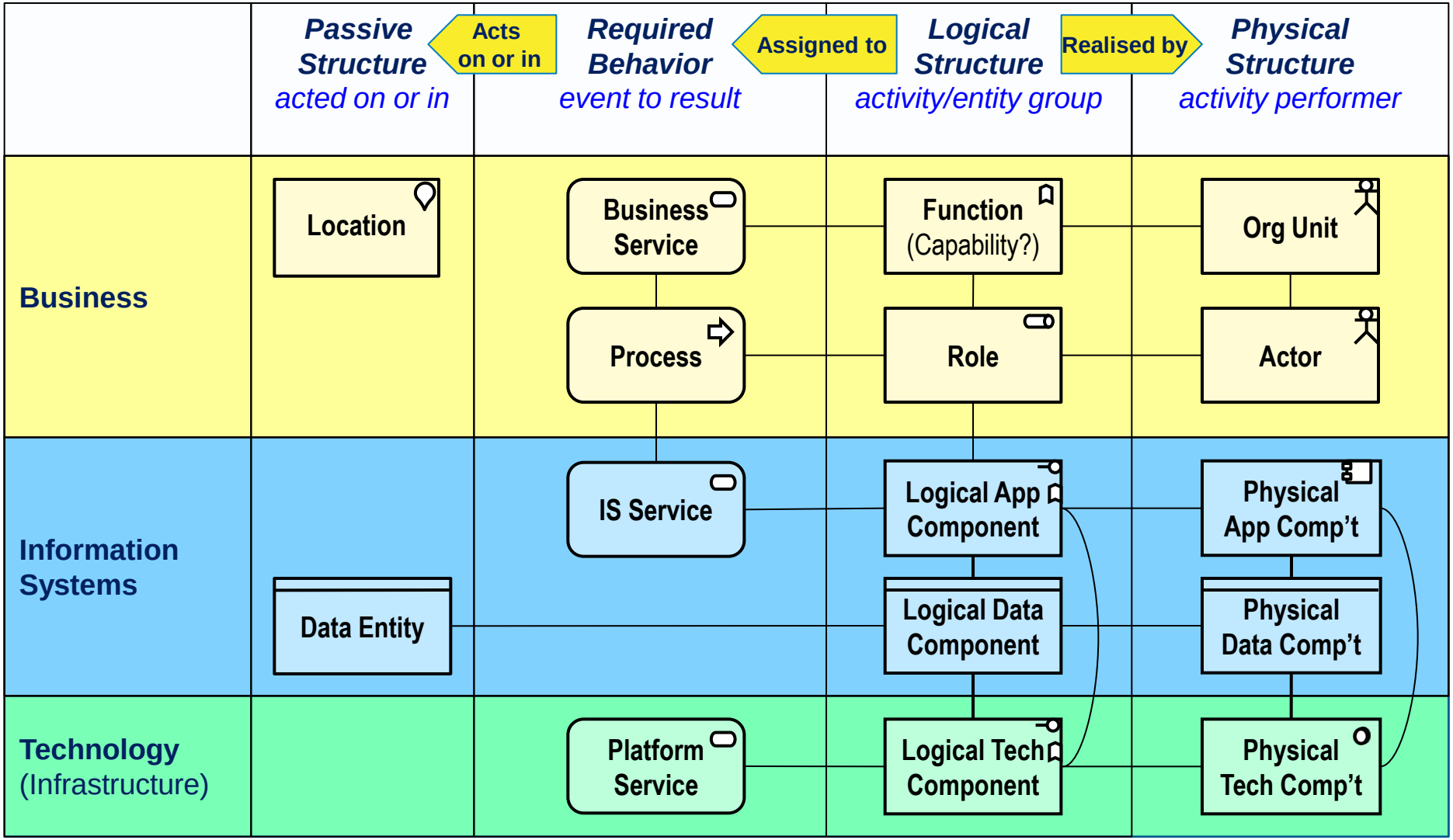
Warning: This slide show bends them both a little

TOGAF separates logical and physical active structures

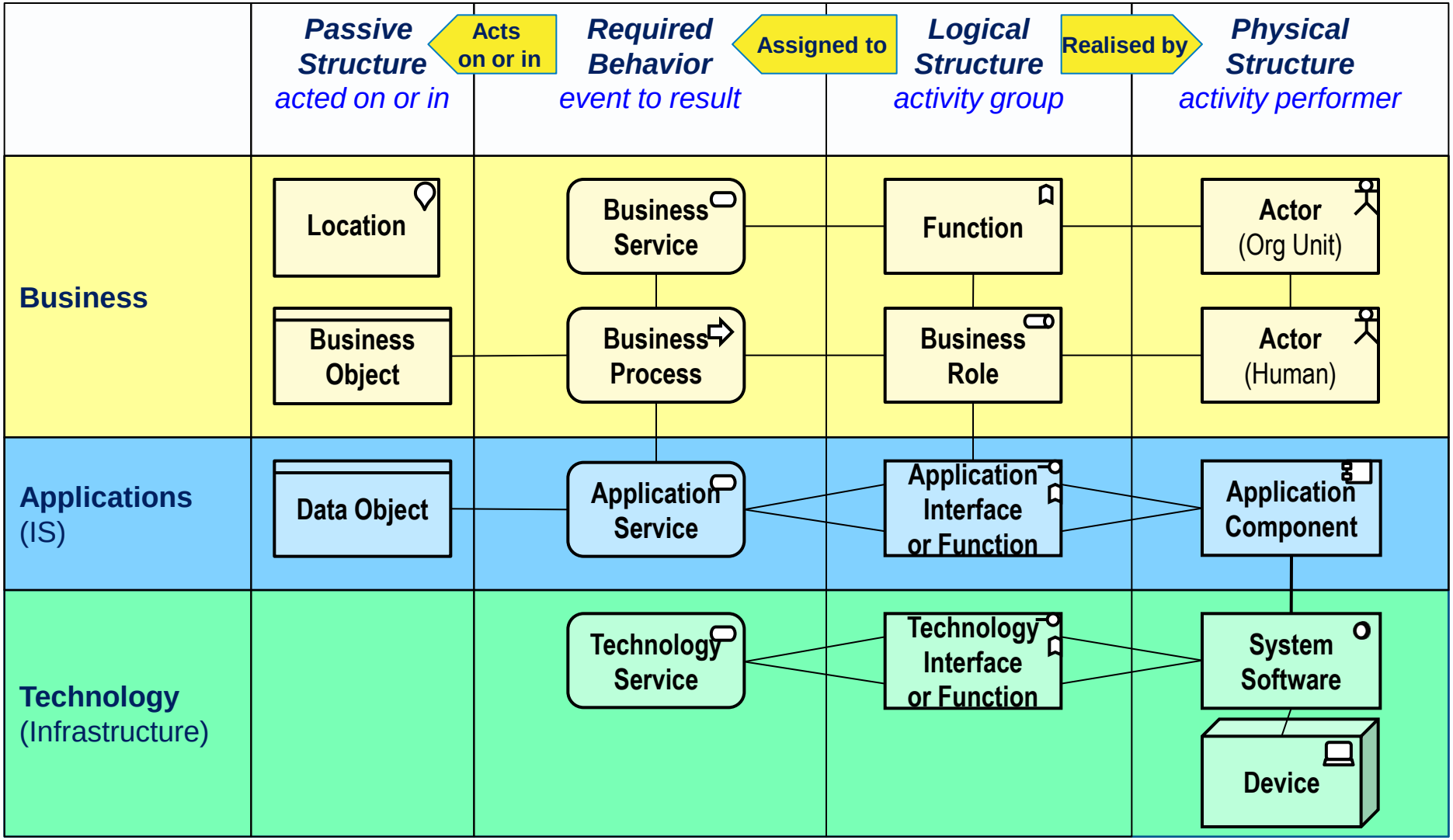
Avancier

	Passive Structure *acted on or in*	***Required Behavior*** *event to result*	***Logical Structure*** *activity/entity group*	***Physical Structure*** *activity performer*
Business			Function (Capability?) Realised by Org Unit Role Realised by Actor	
Information Systems			Logical App Component Realised by Physical App Comp't Logical Data Component Realised by Physical Data Comp't	
Technology (Infrastructure)			Logical Tech Component Realised by Physical Tech Comp't	

TOGAF's core framework – with ArchiMate symbols



ArchiMate's core framework – core entities & associations



Note two small departures from ArchiMate 3.0

Avancier

- We classify **Location** as **passive structure** rather than active structure, and **Function** as **logical structure** rather than behavior.

- We are modelling activity systems
- At the bottom of both behavior and active-structure models are the same atomic activities.
  - Business Processes organise atomic activities in time sequences or dependencies.
  - Business Functions, Organisation Units and Roles organise the *same* atomic activities in what might be called responsibility structures.
- This isn't obvious because *nobody* complete models down to the level of atomic activities.

- Why bother with Business Functions?
- They give us a logical Organisation/Capability structure that saves us from frequent model restructuring.

ArchiMate domains/layers and services

Business Layer depicts

business services
offered to customers, which are realized in the **organization** by
business processes performed by
business actors.

Build a house | Fill a pot hole | Book a train seat

Cut hair | Shampoo hair | Manicure nails

Acknowledge complaint | Resolve complaint | Refund payment

Application Layer depicts

application services
that support the business, and the **applications** that realize them.

Check customer credit | Consolidate drilling reports | Provide weather data

Receive premium | Register claim | Pay claim

Send email | Receive emails | Sort in box

Technology Layer depicts

technology services such as processing, storage, and communication services needed to run the applications, and the computer and communication **hardware** and **system software** that realize those services.

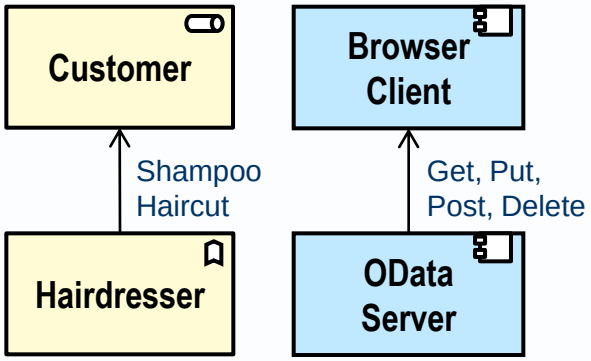
HTTP Get | HTTP Post | HTTP Delete

Encrypt message | Deliver message to receiver | Broadcast message to list

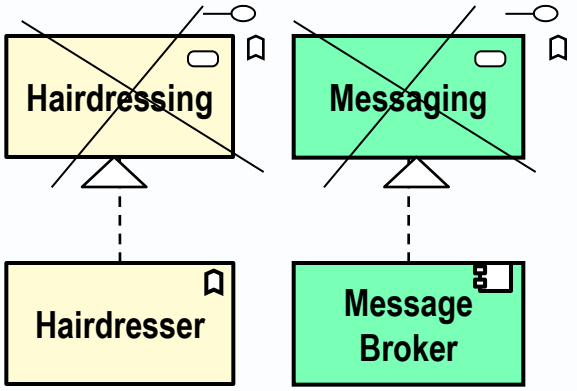
Start transaction | Commit transaction | Roll back transaction

Aside: In-practice advice for ArchiMate diagram drawers

- A service is a *discretely requestable behavior*
- No need to draw a box on a diagram for each service
- You can instead
  - List services in background documentation
  - Annotate a serving arrow with one or more service names
  - Cluster individual services into a function or interface



- Common errors:
  - naming a component or interface as a service
    - don't call it a web service or micro service
    - call it a web app or micro application component
  - defining a *group* of services as a service
    - better call it a function, or assign it for access to an interface.
  - turning a noun into a gerund (a verb acting as a noun)
    - don't say a hairdresser provides a hairdressing service,
    - or a message broker provides a messaging service.



Part 4: TOGAF's entities and artifacts

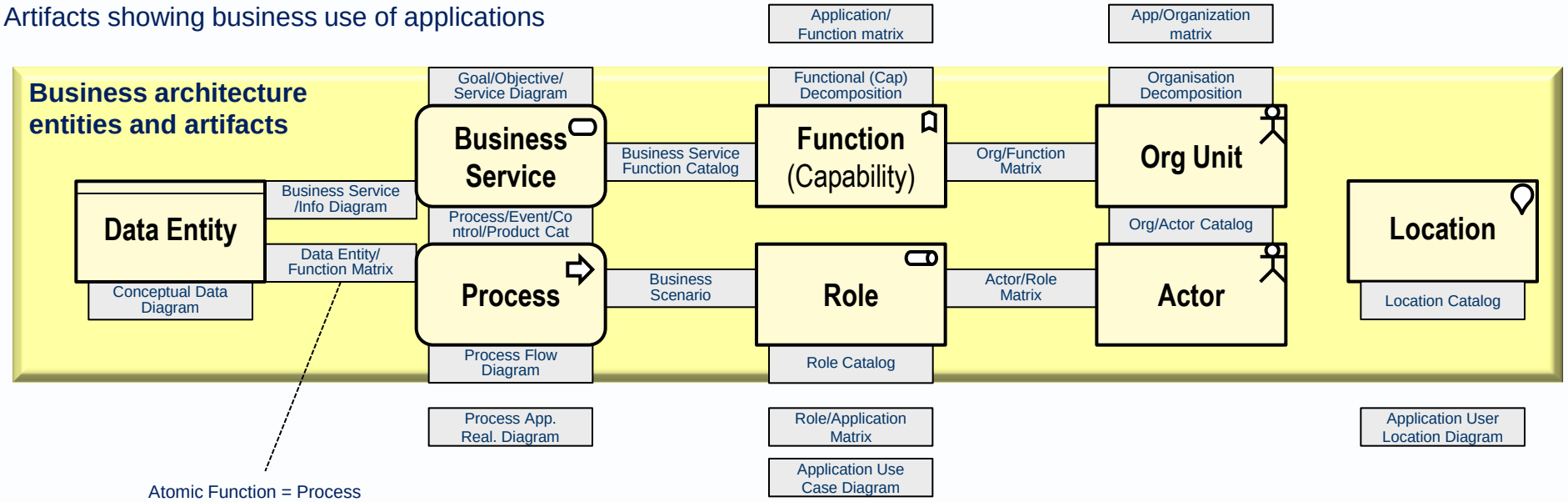
TOGAF' generic metamodel

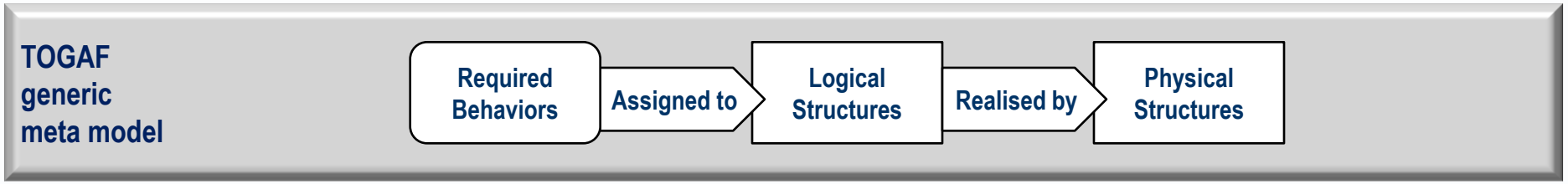
Applied to its metamodel entities and artifacts

TOGAF Business Architecture: entities and artifacts

Avancier

Artifacts showing business use of applications

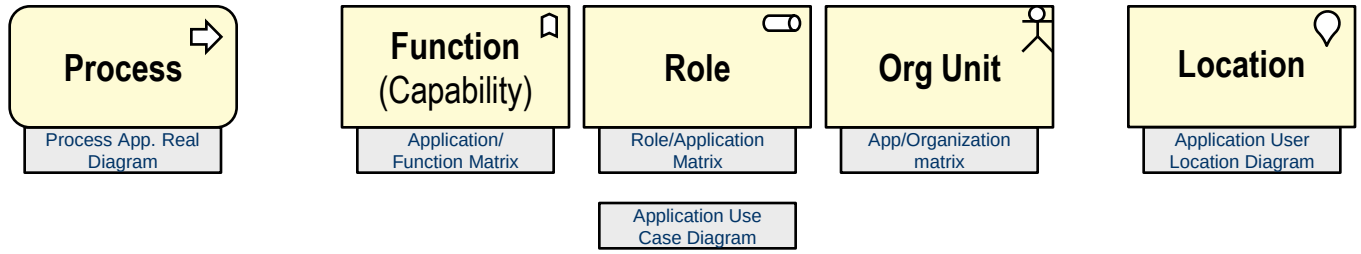




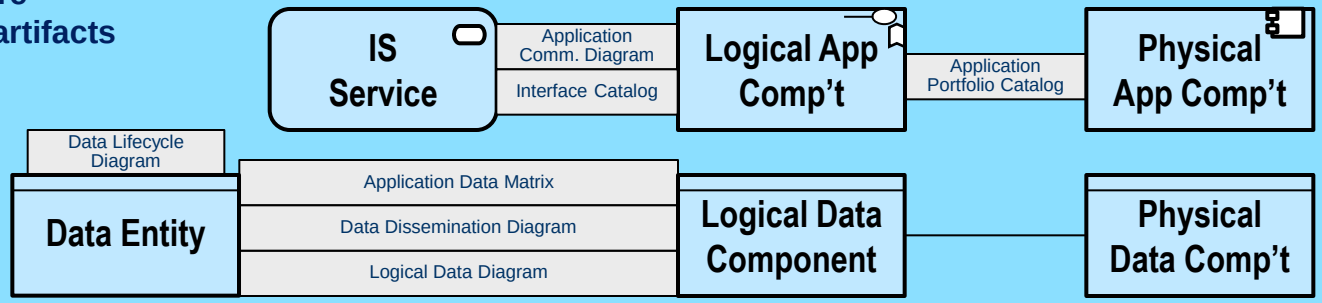
TOGAF Information Systems: entities and artifacts

Avancier

Artifacts showing
business use of applications

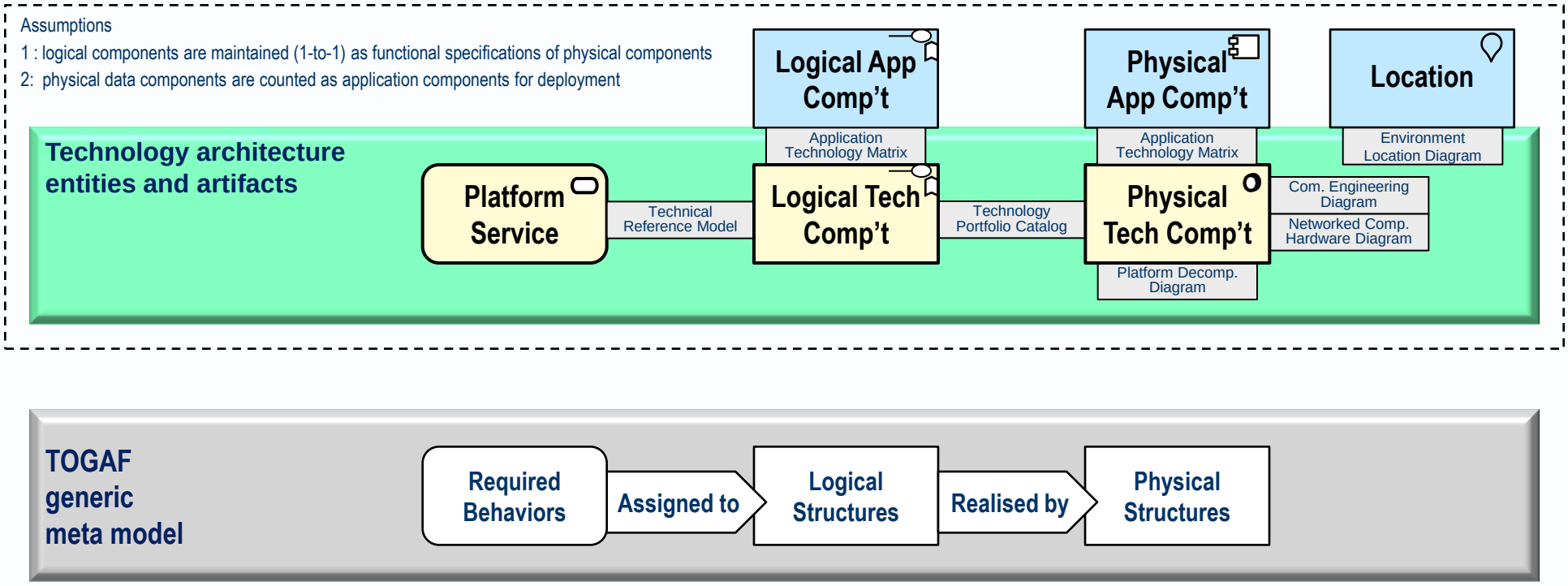


IS architecture entities and artifacts

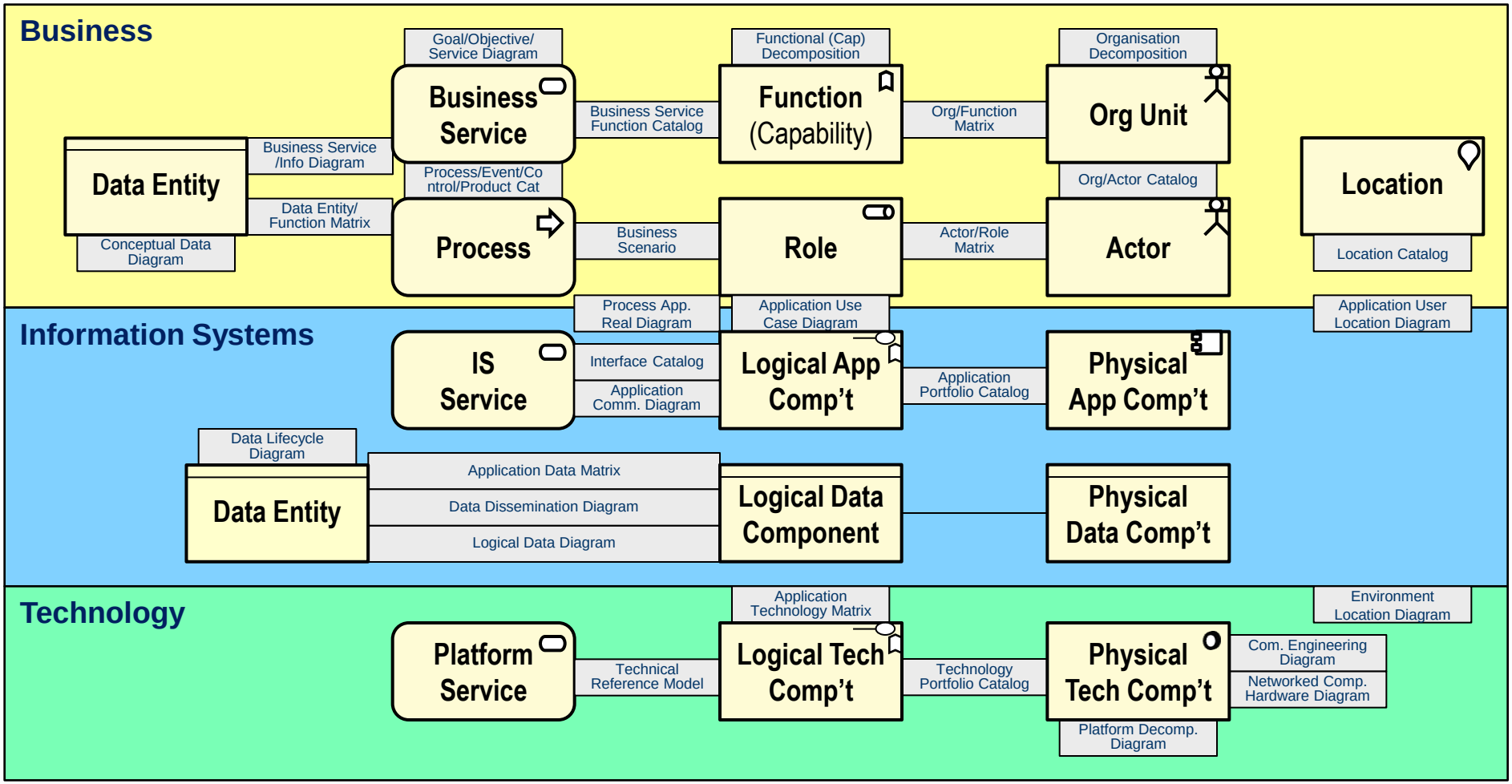


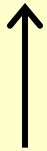
TOGAF generic meta model

TOGAF Technology Architecture: entities and artifacts



Summary outline: architecture entities and artifacts



Delegator	Composition	Generalisation	Idealisation
			
Servant	**Decomposition**	**Specialisation**	**Realisation**

Part 5: More about abstraction

Aligning **ArchiMate®** with **TOGAF®**

Warning: This slide show bends them both a little

Four kinds of abstraction in TOGAF and ArchiMate

Avancier

Abstraction in ArchiMate - relationship symbols

Delegator	Composition	Generalisation	Idealisation
Client Server and Client Server	Coarse-grained composite Mid-grained composite Fine-grained composite Elementary part	Universal Fairly generic Fairly specific Unique	Conceptual Logical Physical Real
Servant	Decomposition	Specialisation	Realisation

Uses of abstraction in TOGAF

Enterprise Continuum

Delegator	**Composition**	**Generalisation**	**Idealisation**
Business **Applications** **Technologies**	**Enterprise/Strategy** **Segment** **Capability**	**Foundation** **Common System** **Industry** **Organisation**	**Requirements** **Architecture continuum** **Solution continuum** **Deployed solutions**
Servant	**Decomposition**	**Specialisation**	**Realisation**

TOGAF Enterprise Continuum with quotes and references

Avancier

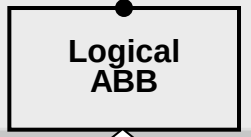
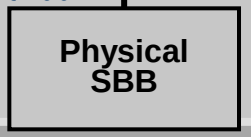
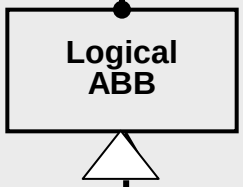
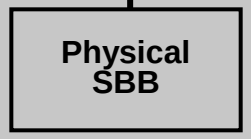
Enterprise Continuum level	Logical - Physical	ABB - SBB
Requirements & context 		
Architecture Continuum "a repository of architectural elements…" 	**"Logical:** An implementation-independent definition of the architecture, often grouping related physical entities according to their purpose and structure."	**"Architecture Building Block (ABB):** a constituent of the architecture model." "Architecture Building Blocks: Architecture documentation and models…"
Solutions Continuum "contains implementations of… definitions in the Architecture Continuum." 	**"Physical:** A description of a real-world entity. Physical elements in an enterprise architecture may still be considerably abstracted from Solution Architecture, design, or implementation views."	**"Solution Building Block (SBB) :** a candidate physical solution for an Architecture Building Block (ABB)" "Solution Building Blocks: Implementation-specific building blocks…"
Deployed Solutions		SBBs deployed as live solutions

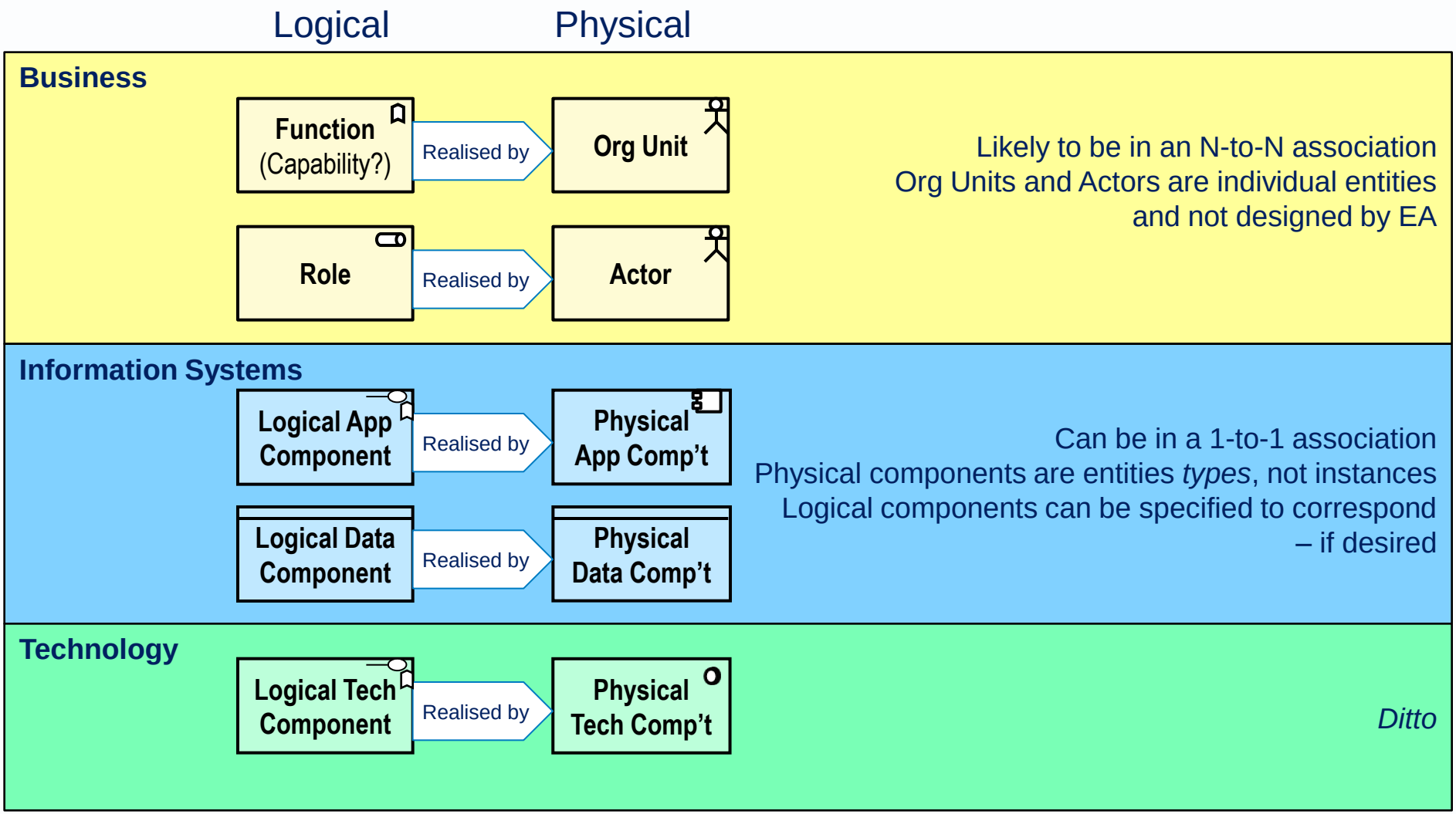
Figure 2-3 Enterprise Continuum
Figure 6-2 Management Frameworks to Co-ordinate with TOGAF
Figure 6-3 Interoperability and Relationships between Management Frameworks
Figure 28-2 Consolidated Gaps, Solutions, and Dependencies Matrix
Figure 40-1 Summary Classification Model for Architecture Landscapes AND Figure 40-2 Summary Classification Model for Solutions.

TOGAF Enterprise Continuum mapped to ArchiMate

Avancier

Enterprise Continuum	Generic meta model	ArchiMate
Requirements & context 	**Required behaviors** Discretely requestable behaviours – defined in the Architecture Requirements Specification	**Service**
Architecture Continuum **“**a repository of architectural elements…” 	**Logical Structures** “Architecture building blocks" (ABBs) Vendor neutral. Ideal or potential components. Defined by services delivered , activities performed , data maintained.	**Role, Interface, Function.**
Solutions Continuum **“**contains implementations of… definitions in the Architecture Continuum.” 	**Physical Structures** “Solution building block" (SBBs) Vendor or technology specific. Component types chosen to perform activities and implement logical structure(s)	**Component, Node.**
Deployed Solutions	Deployed instances.	**Actor**

Associations between logical and physical building blocks



Capability?

- Capability may be equated with high-level function.
  - Functional and capability decomposition hierarchies are indistinguishable.
  - "Capabilities are typically expressed in general and high-level terms and typically require a combination of organization, people, processes, and technology to achieve. For example, marketing, customer contact, or outbound telemarketing." TOGAF 9.1
- Or capability may be seen as function + goals + resources needed to realise the function, which is to say a system.

- *"Architecture' has two meanings depending upon the context:*
  - *1. A formal description of a system...*
  - *2. The structure of components, their inter-relationships..."* (TOGAF 9.1)
- Can surely be rephrased with no loss of meaning:
  - *1. A formal description of a capability...*
  - *2. The structure of building blocks, their inter-relationships...*

Mainstream EA takes a particular view of a business system

- A collection of
  - actors (structures) will perform roles in
  - activities (behaviours) to provide required
  - services.
- The system
  - changes externally when service types are changed
  - changes internally when activity types or actors' roles are changed.
  - does not change when individual actors are replaced.
- An abstract description/model of the system should be
  - accepted by its sponsors/key stakeholders
  - revised and approved before the operational system is changed.

More about Architecture Terms and Concepts?

- See the papers on the home page at http://avancier.website.