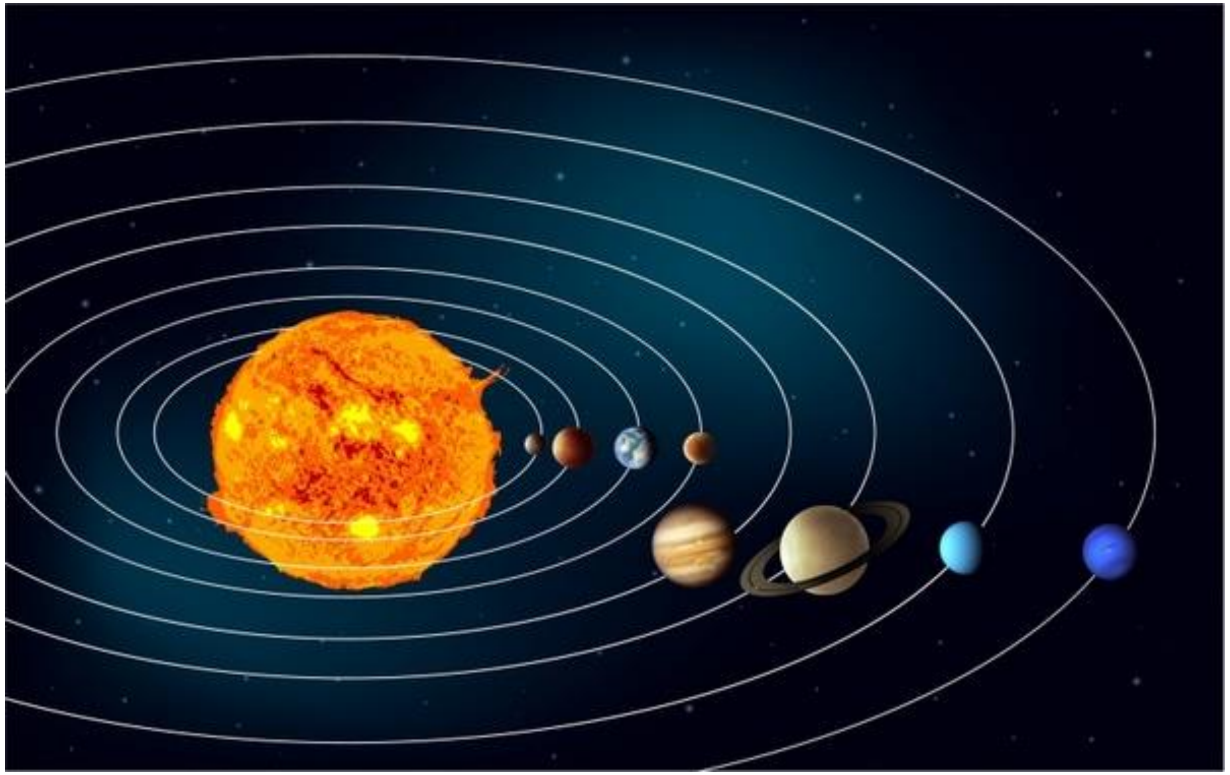
System Concepts



that underpin enterprise and business architecture

Last updated: 13/07/2020

An extract from training at http://avancier.website

This slide show

- “EA regards a business as a system of systems” TOGAF
- But what is a system?

- This slide show introduces
  - A cybernetician’s view of systems – after Ashby
  - A general view of dynamic activity systems – after Meadows
  - Principles of enterprise and business architecture

- And relates them to the meta models of ArchiMate and TOGAF

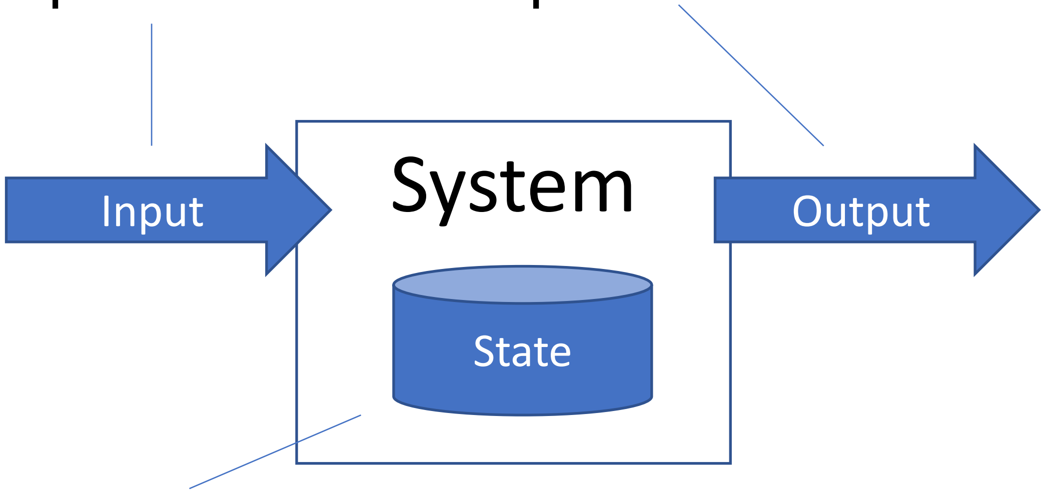
A cybernetician’s view of systems (after Ashby)

Last updated: 25/05/2020

An extract from training at http://avancier.website

The systems of interest

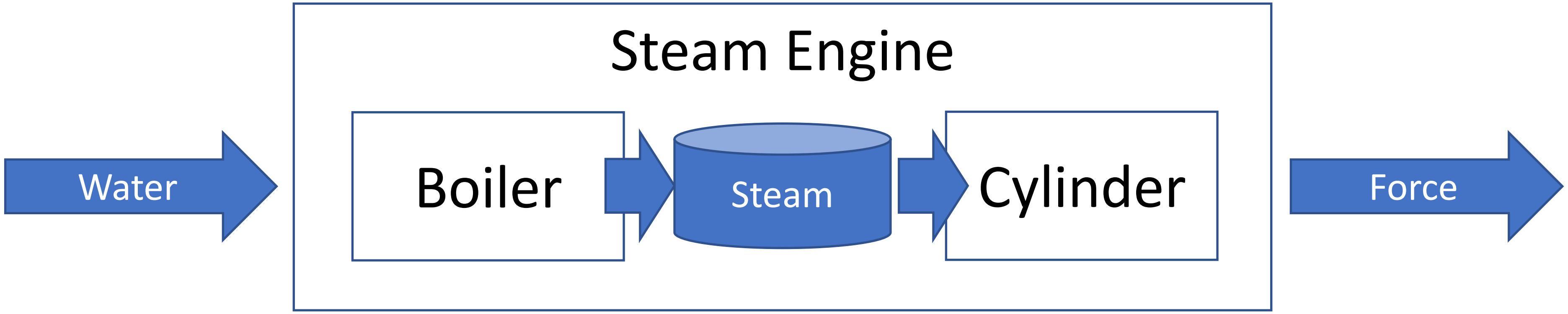
- Cybernetics is about systems that
- Transform input flows into output flows



- And advance internal state variables

System composition

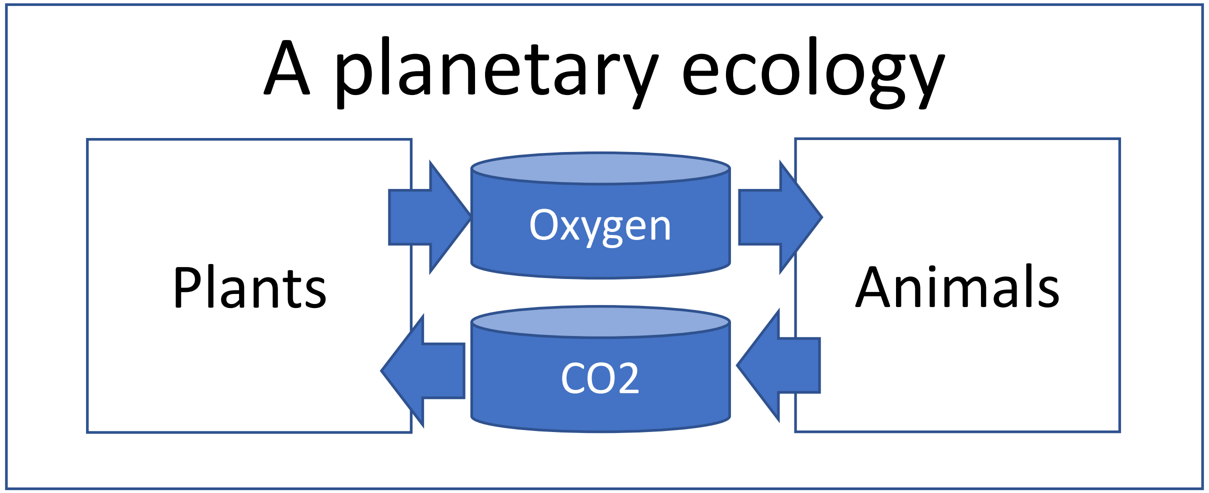
You can connect two systems as subsystems of (actors in) a wider system



In EA, there is usually a large and complex network of systems

Closed systems

- This ecological system's state includes the volumes of oxygen and CO2 in the atmosphere and the biomasses of plants and animals


A planetary ecology

Plants

Oxygen

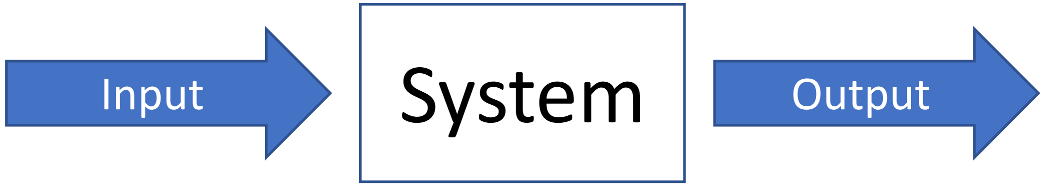
CO2

Animals

- In EA, we usually model *open* systems, which connect to a wider environment

Open systems, in a wider environment

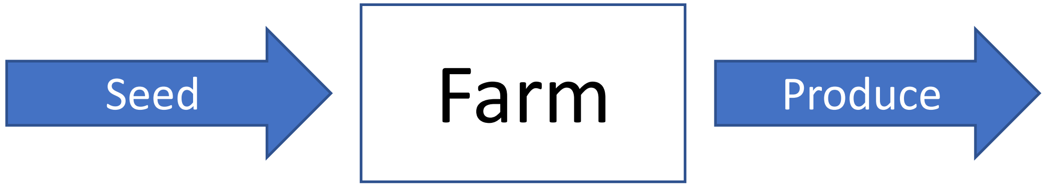
- An open system is driven by inputs from its environment.
  - Forces (e.g. gravity, mechanical push/pull)
  - Energy (e.g. light, sound, electrical current)
  - Matter/materials (e.g. water, widgets, cans of soup)
  - Information (e.g. messages carrying data structures).



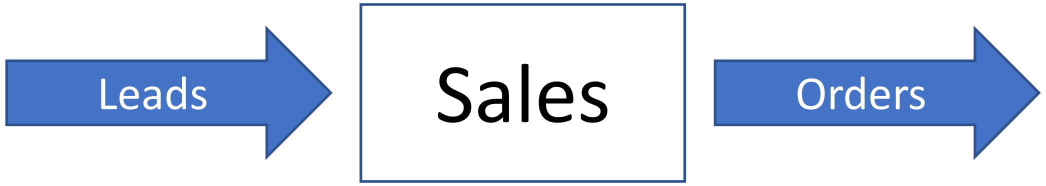
- In EA, the concern is mostly *data flows*
  - A data flow encodes information in some form of matter and/or energy
  - An energy supply, and the laws of thermodynamics, are taken for granted

Defining a system’s boundary

- You may draw a boundary around a physical entity
  - such as a farm, a factory, a shipyard, or some part thereof.

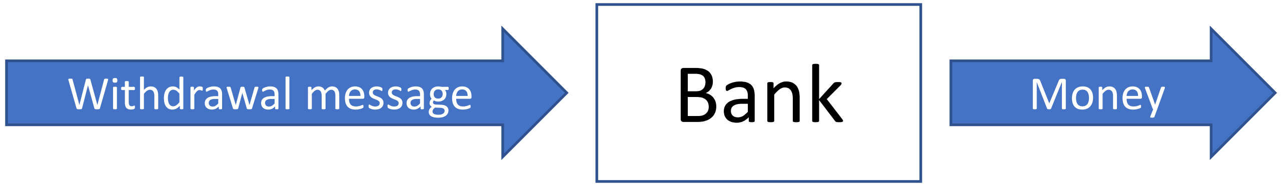


- In EA, we usually draw a legal or logical boundary around a social entity
  - actors distributed in space and connected by information flows.



The causes and effects of behavior

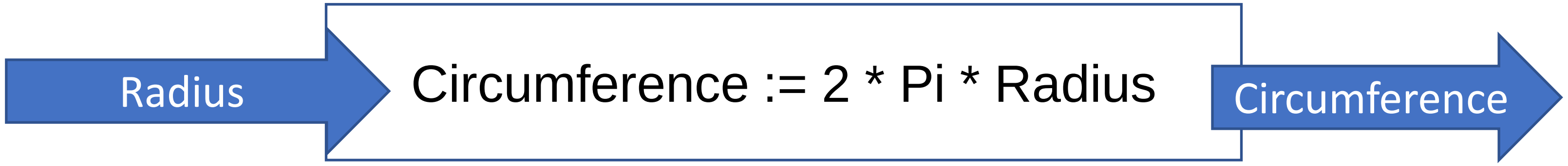
- In the real world, things happen for which no cause is described.
- A bank does not know why a customer wants to withdraw money



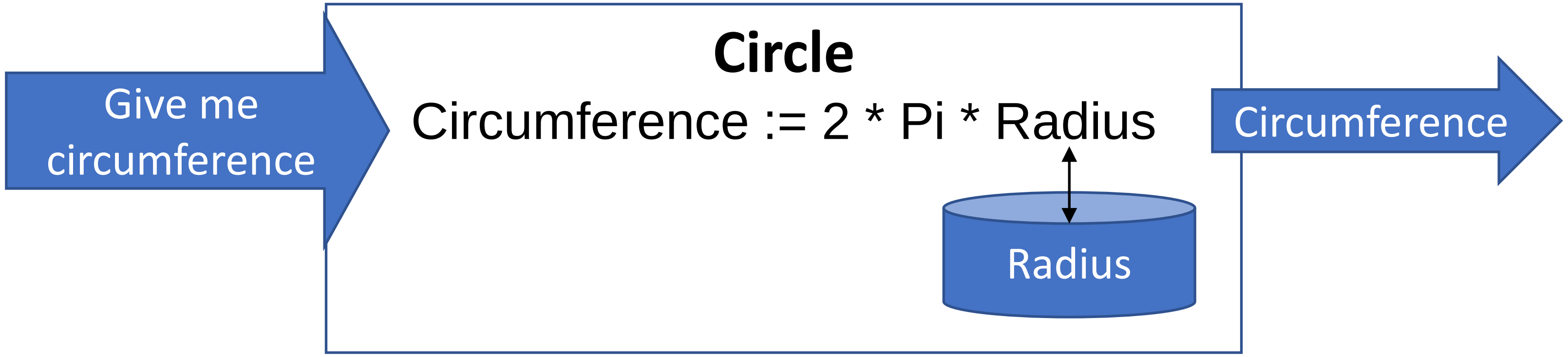
- In EA, whatever happens *within* a business system is traceable to a describable cause - such as a cash withdrawal event, or a time event

What prompts behaviors to produce output?

1. Input only (called a "function" in UML)



2. Input with reference to an object's persistent state/memory

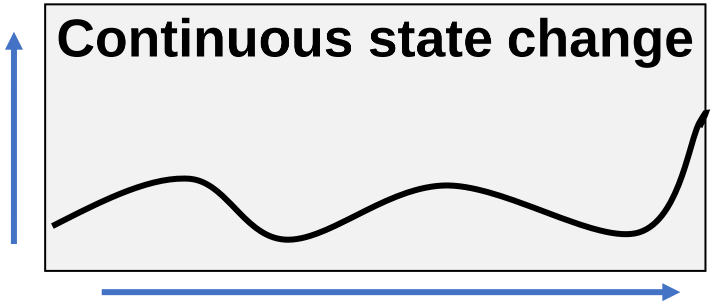


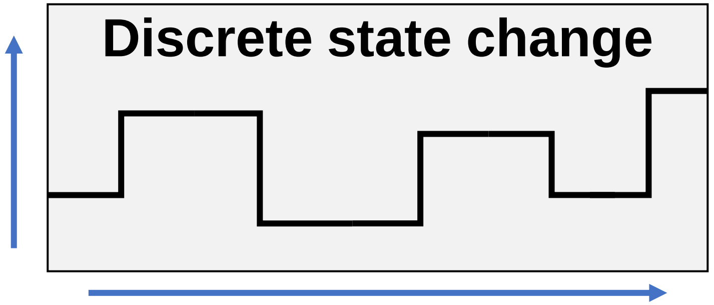
3. An internal state-change (in turn caused by an input or time event)

An extract from training at http://avancier.website

Continuous and discrete dynamics

- Business systems are dynamic in that they change state over time
- Modellers often convert continuous dynamics to discrete





- In EA, most business systems are **discrete event-driven systems**
- They respond to discrete events - and change state in discrete steps

In enterprise and business architecture

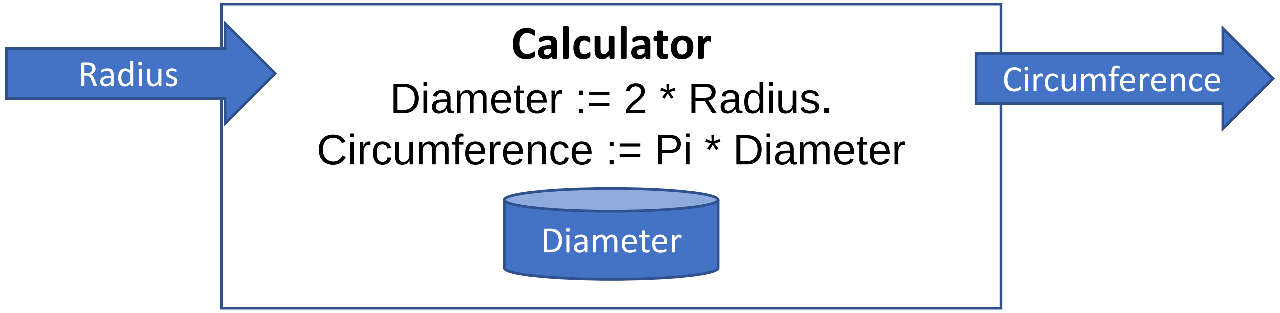
- Behaviors are triggered by discrete events

- Behaviors are deterministic
  - though human actors are assumed to be self-willed
  - and choices between optional behaviors might be probabilistic

- Behaviors refer to system state - and may change or report it

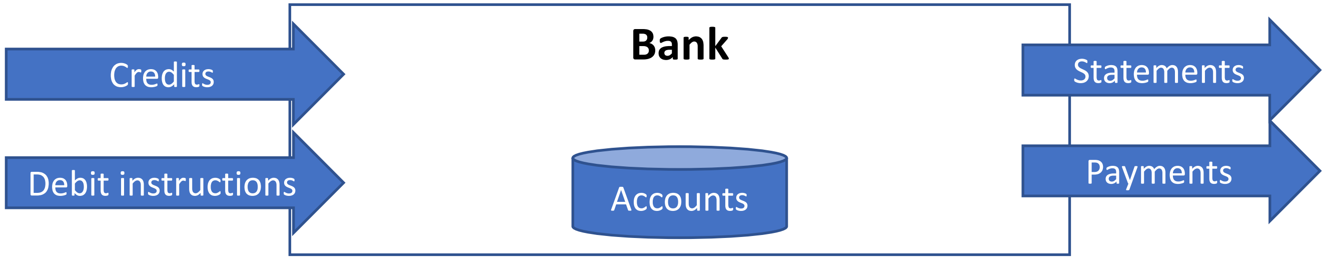
Stateful behavior

- A “stateless” system does not remember past events or their effects
- In EA, most business systems retain both short-term memories



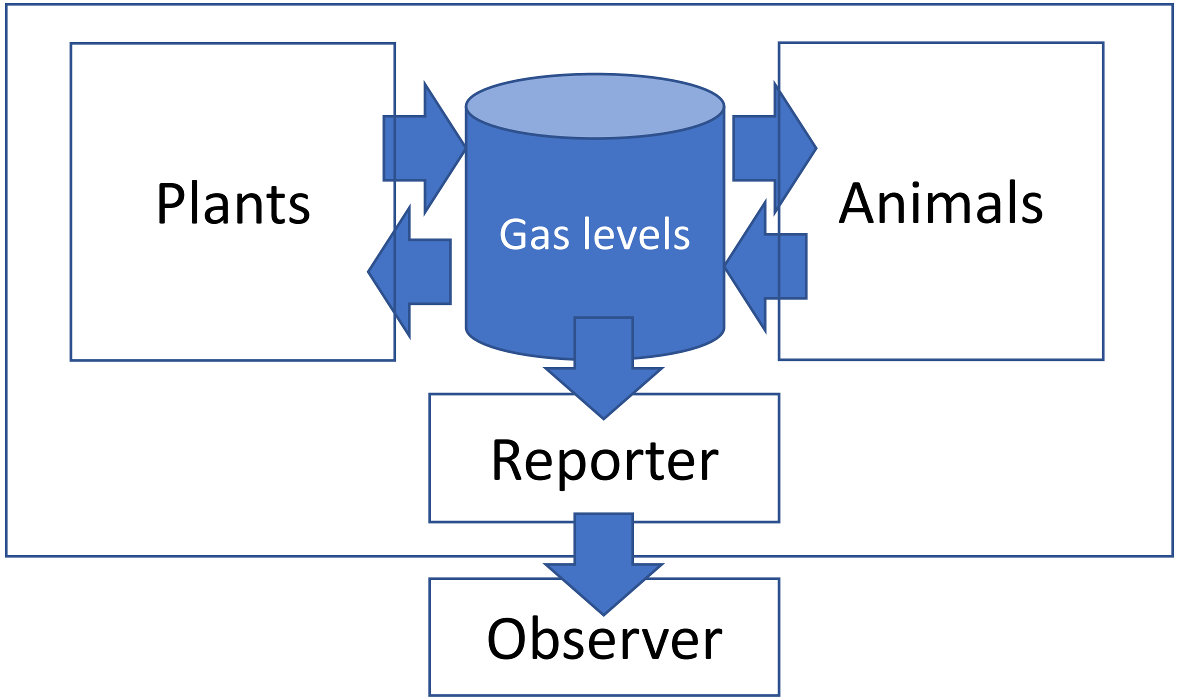
Diameter := 2 * Radius.
Circumference := Pi * Diameter

- And long-term memories in data stores



Encapsulation of state

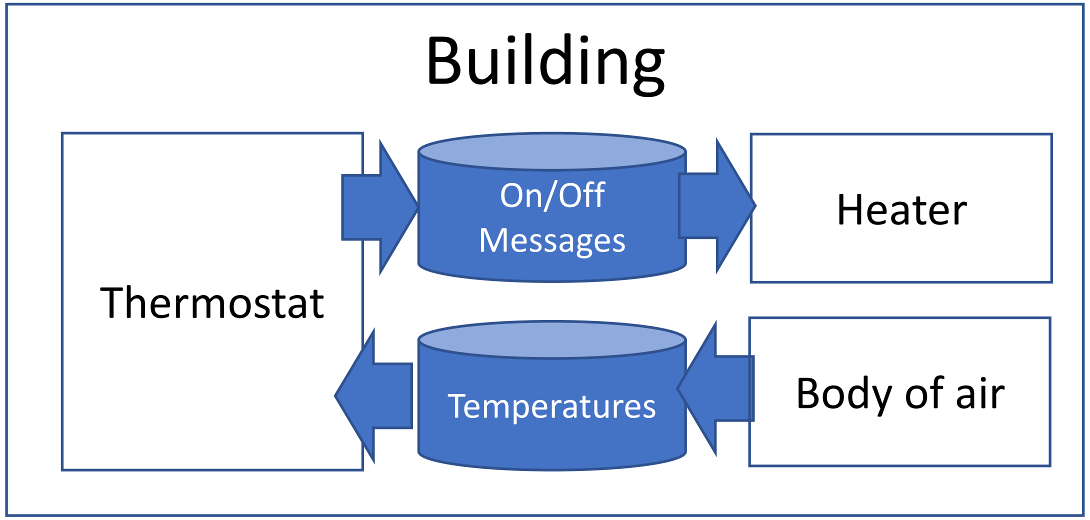
- We may speak of a system’s state as an internal structure or memory

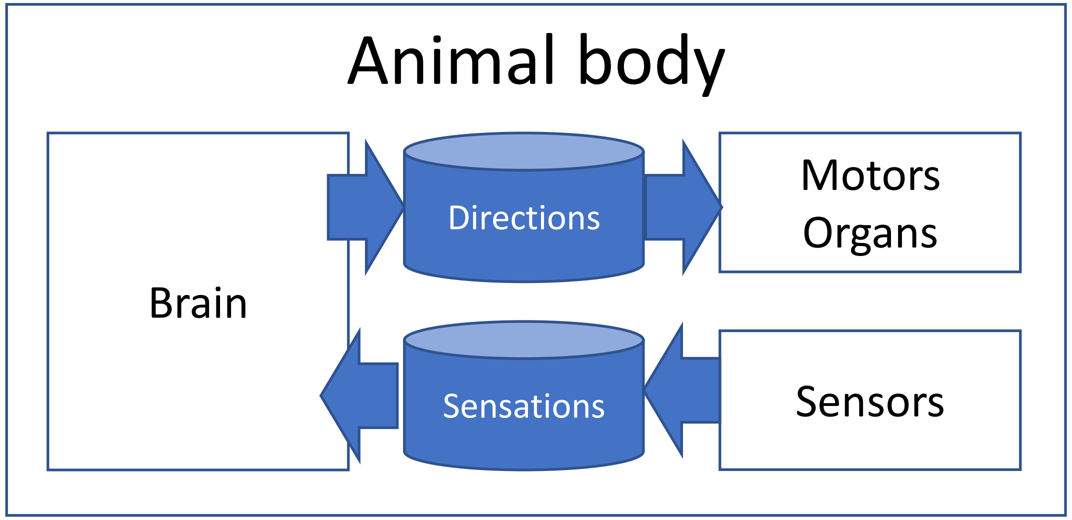


- In EA, observers can inspect that internal state

Cybernetic feedback loops

- Weiner introduced cybernetics as the science of control in machines and organisms
- A control system senses a target system’s state, and sends messages to change that state





- In EA, information feedback loops are commonplace

From cybernetics to activity system theory

Ashby’s “Design for a Brain” 1952

Cybernetic design for a brain	General activity system
Brain cells	Actors
interact in processes to	interact in orderly activities to
maintain body state variables by	maintain system state and/or
receiving/sending information	consume/deliver inputs/outputs
from/to bodily organs/sensors/motors.	from/to the wider environment.

Activity system concepts (after Meadows)

Last updated: 25/05/2020

An extract from training at http://avancier.website

What characterises an activity system?

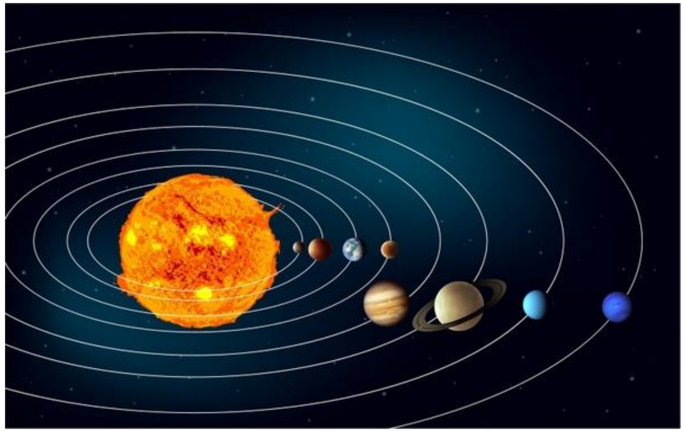
- In an activity system
  - *actors* interact in
  - regular *activities*

Actor

Planet

Activity

Orbit



- And in a *designed* system the activities are designed with *aims* in mind

Aim

What are actors and activities?

- Actors (people, planets, cells, molecules…) interact to perform the characteristic activities of system.

- An **actor** is a structure (or continuant) that is able to perform activities
  - An individual component that is made, bought, hired or employed

- An **activity** is a behavior (or occurrent) that changes or makes something
  - a regular process that advances the state of the system
  - repetition produces a **line of behavior** (a trajectory of state changes over time).

What are aims or purposes? (after Meadows)

- **Actors** occupy space, and may be visible or tangible
- **Activities**, which run over time, are harder to see,
- **Aims** are even harder.

- The aims of a system may be seen as
  1. intentions or goals – of external and/or internal actors, or
  2. what the system does – its effects by way of producing outputs and/or advancing system state (Meadows' view)

- In EA, it is presumed that 2 is designed to meet 1

Is every social entity an activity system? (after Meadows)

- A system isn’t just any old collection of **actors** who communicate
- It is a collection of **actors** organized to perform particular **activities**
- The system is characterized by its **activities** in response to inputs

- In EA, changes to the roles of actors in a business are often addressed by a business change function working in parallel with the EA function

Is there anything that is not an activity system? (after Meadows)

- Yes, a **passive structure**
  - Linnean classification of species
  - Dewey decimal system
  - Periodic table in chemistry
  - Other hierarchical and tabular structures

- Also, an informal **social entity**, a group of actors who do not interact in the particular ways that characterize a system.
  - (And grouping actors into units under a management hierarchy is not enough to call that “organization” a system.)

How to know you are looking at a system? (after Meadows)

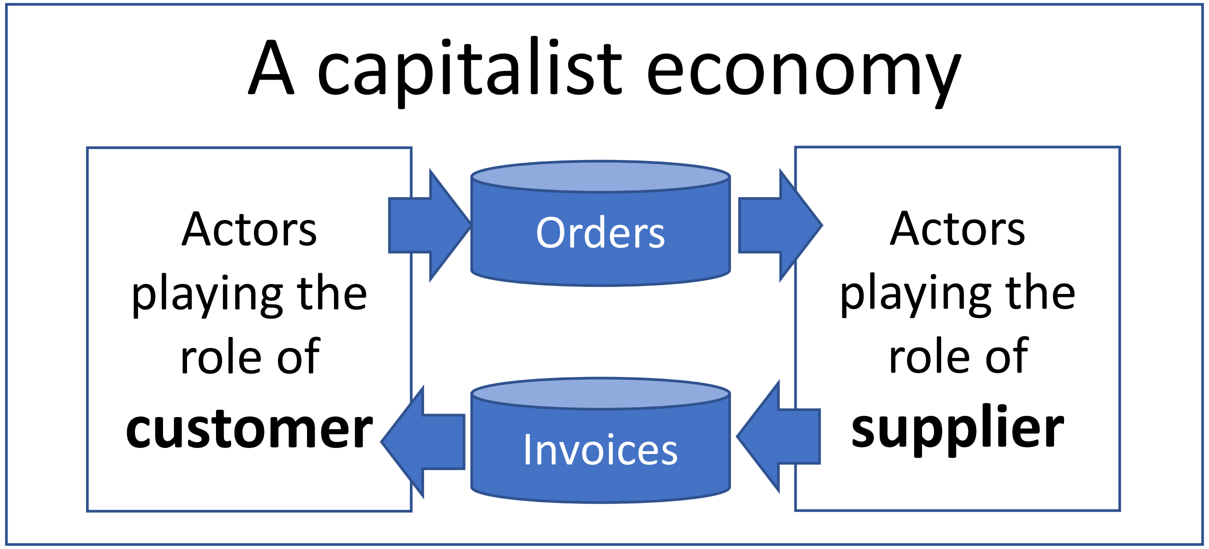
And not just a bunch of actors or stuff happening?

1. Can you identify *roles* played by **actors** in interactions?
2. Do **actors** cooperate in those roles to produce effects (change the state of the system and/or produce outputs)?
3. Do those effects differ from the effects of **actors** on their own?
4. Are the **activities** regular and repeatable?

In EA, we usually model *roles* rather than **actors**.

Modelling roles rather than actors

- In business systems, actors exchange *messages*
- And may retain *memories* of messages sent and received


A capitalist economy
Actors playing the role of **customer**
Orders
Invoices
Actors playing the role of **supplier**

- In EA, we usually model Roles rather than Actors

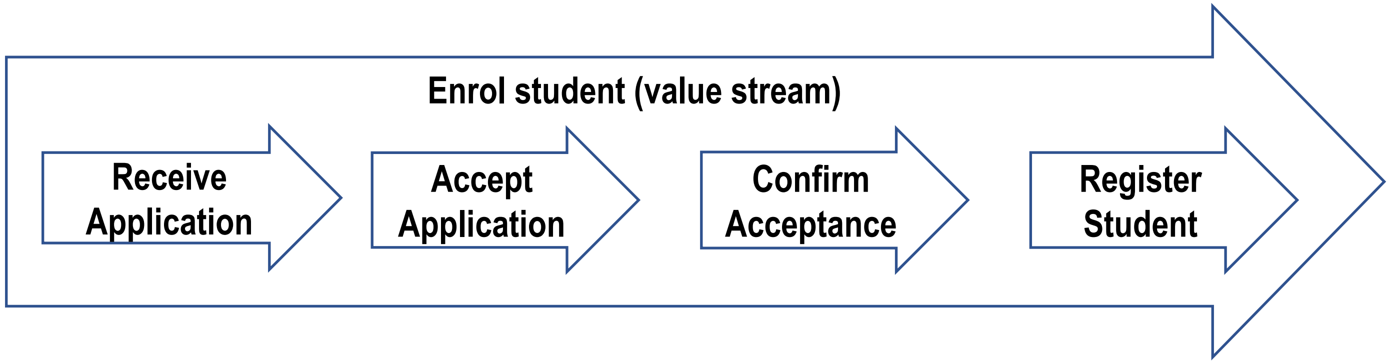
Imposing hierarchies on networks and social entities

Last updated: 25/05/2020

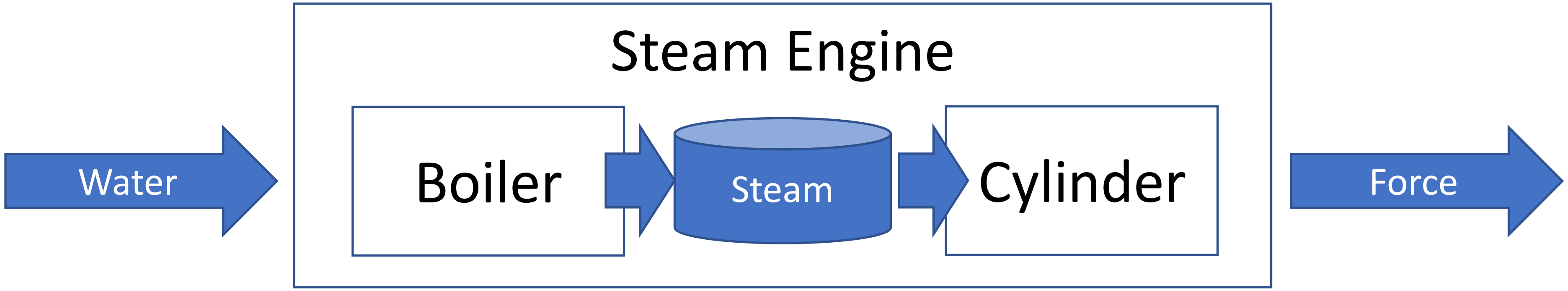
An extract from training at http://avancier.website

Behavioral decomposition v. structural decomposition

- You can decompose long activities into *shorter* activities.



- You can decompose large actors into *smaller* actors.



An extract from training at http://avancier.website

How to make sense of a hugely complex business?

- A large business is a vastly complicated *network* of

Actor actors (subsystems, components) interacting in a

Activity activities (value streams, processes) to meet a

Aim aims (goals or objectives)

- How to make sense of all that? make it manageable?

Imposing hierarchies on networks

- You can decompose

Actor large subsystems into finer-grained actors (though human actors are atomic),

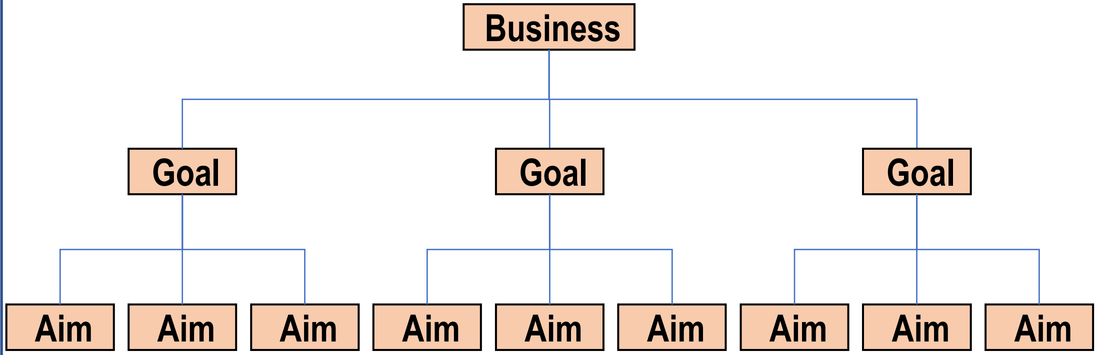
Activity long processes into finer-grained activities

Aim strategic goals into more tactcal objectives or aims

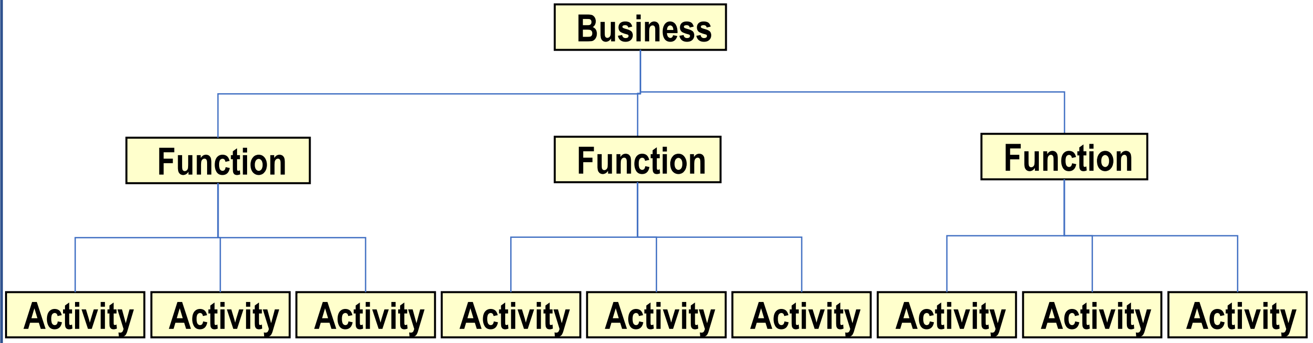
- You can decompose or compose them successively to form a hierarchy
- Typically 3 or 4 levels deep, but potentially deeper.

These hierarchies are views (not systems) that people draw to help them understand and manage social entities

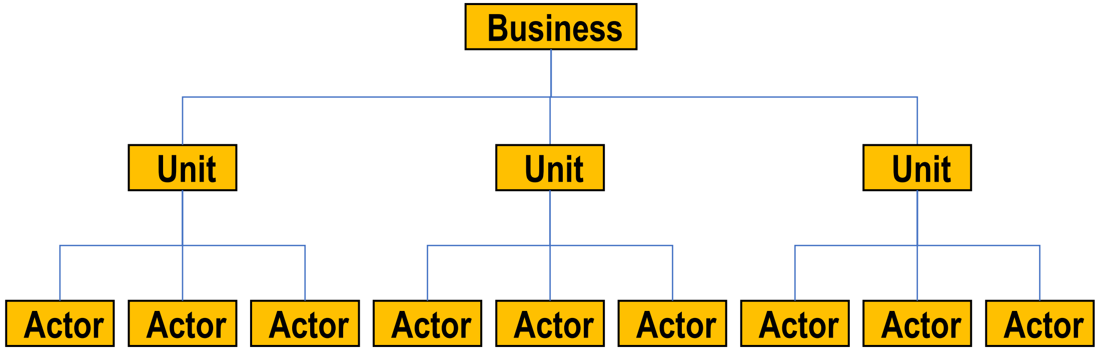
Goal cascade



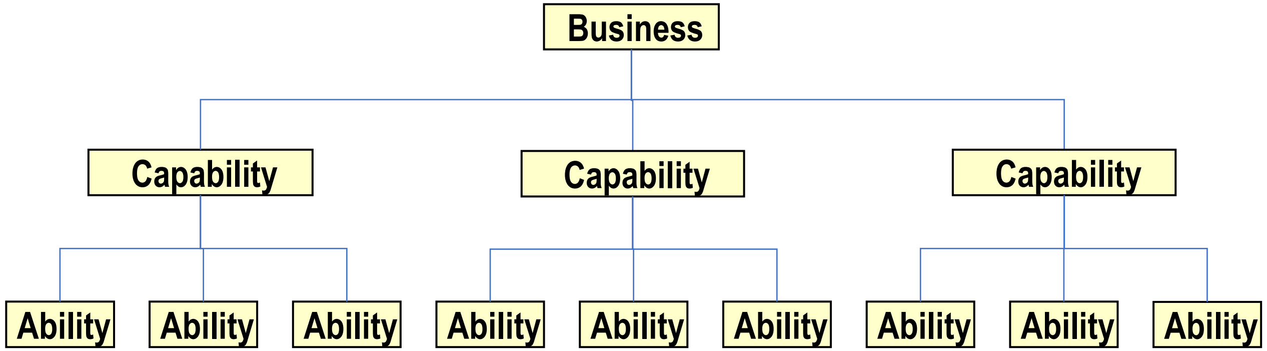
Functional decomposition



Organization (management structure)



Capability map (some match capabilities to goals or functions)



A Function or Capability hierarchy can be used to show a simple overview of a whole business, and categorise other things.

When we speak of the Capability to meet a Goal, perform a Function, deliver a Service, complete a Process or Project, we are speaking of the system to do it.

An extract from training at http://avancier.website

Note

- Every attempt to impose a hierarchical structure on a business
  - (on its networks of aims, actors and activities or abilities)
- is an artificial compromise.

- Different stakeholders may prefer different hierarchical "maps".

Principles of enterprise and business architecture

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

EA as business system planning

- “EA regards a business as a system of systems” TOGAF

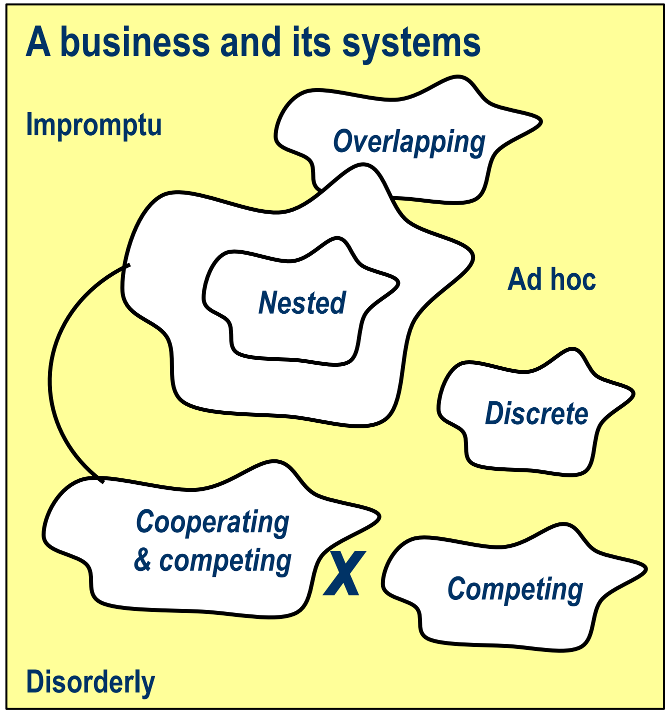
- EAs should understand and plan changes to business systems
  - in which regular activities create and use business data
  - in the provision of business services/products
- (They cannot model ad hoc, unsystematic, activities.)

- Note that an enterprise’s business planning sits above its business system planning, as discussed here https://bit.ly/2OIJVqr

Focusing on the systems 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.

- In EA, though architects govern the whole portfolio of systems, a change project usually focuses attention on a “system of interest", or a relationship between systems.



Core business architecture concepts

- A business features **actors** that perform regular **activities** to meet **aims**
  1. A target **aim** for **activities** = a **Goal** or **Objective**
  2. A contract for **activities** that yield a result (or product) of value to an external actor = a **Service**
  3. **Activities** sequenced to deliver a service = a **Process**
  4. **Activities** grouped for understanding = a **Function**
  5. **Activities** grouped for management = an **Organization unit**
  6. **Activities** grouped for assignment to one or more actors = a **Role**
  7. An individual that plays one or more roles = an **Actor**
  8. Information encoded in a message / memory = a **Data Flow** / **Data Store**

Capabilities are associated 1-to-1 with whichever of the above you choose.
Products are defined in Service contracts
Do you need Function? Actor? Can you combine Service/Process?

“A use case… yields an observable result.. of value for … stakeholders” UML

Business processes

- A business process can yield two kinds of observable result:
  - a) internal state changes – recorded in data
  - b) external outputs - data, sometimes associated with goods.
- The value of a result to its user can be declared in a service contract for the process.
- The user of a result may be an internal actor or an external actor/customer.

- Where the user is a customer, some call the process a value stream.
  - And a value stream diagram is a high-level informal representation of the process.
- However, this use of the term depends on where you draw the system boundary
  - Since one person's external customer is another person's internal actor.

Business services

- Seen from the outside, a designed activity system provides **services** to external actors

- A **service** is a discrete behavior – as seen from outside the system of interest
  - It encapsulates a process, value stream or use case
  - It yields a result of value to a user/consumer/customer
- E.g.
  - "Replace tyre" gives a customer a new tyre.
  - "Polish shoes" gives a customer shiny shoes.
  - "Book train ticket" gives a customer a paper ticket and/or a digital ticket
  - “Turn data into Insights” gives a managers insights into their business

Service contracts

- In logic, a behavioral assertion
  - Precondition <Process> Post condition.

- A **service contract**
  - Service name
  - Entry conditions
    - Inputs and other preconditions
  - Exit conditions (results of value to users/consumers/customers)
    - Outputs: information, goods
    - Internal state changes
  - Non-functional qualities
    - Speed, volume, availability, security etc.

A meta process: to define a business activity system

1. Define **aims** or **goals**
2. Define **activities** to meet goals
   - externally in the form of **service** contracts
   - internally as **process** flows
3. Define **data** created and used by **activities**
   - Physical flows of energy or materials are only modelled if they carry data
4. Define **roles** in which **actors** perform **activities**
   - Physical actors (person, printer or power station) are only modelled in such roles

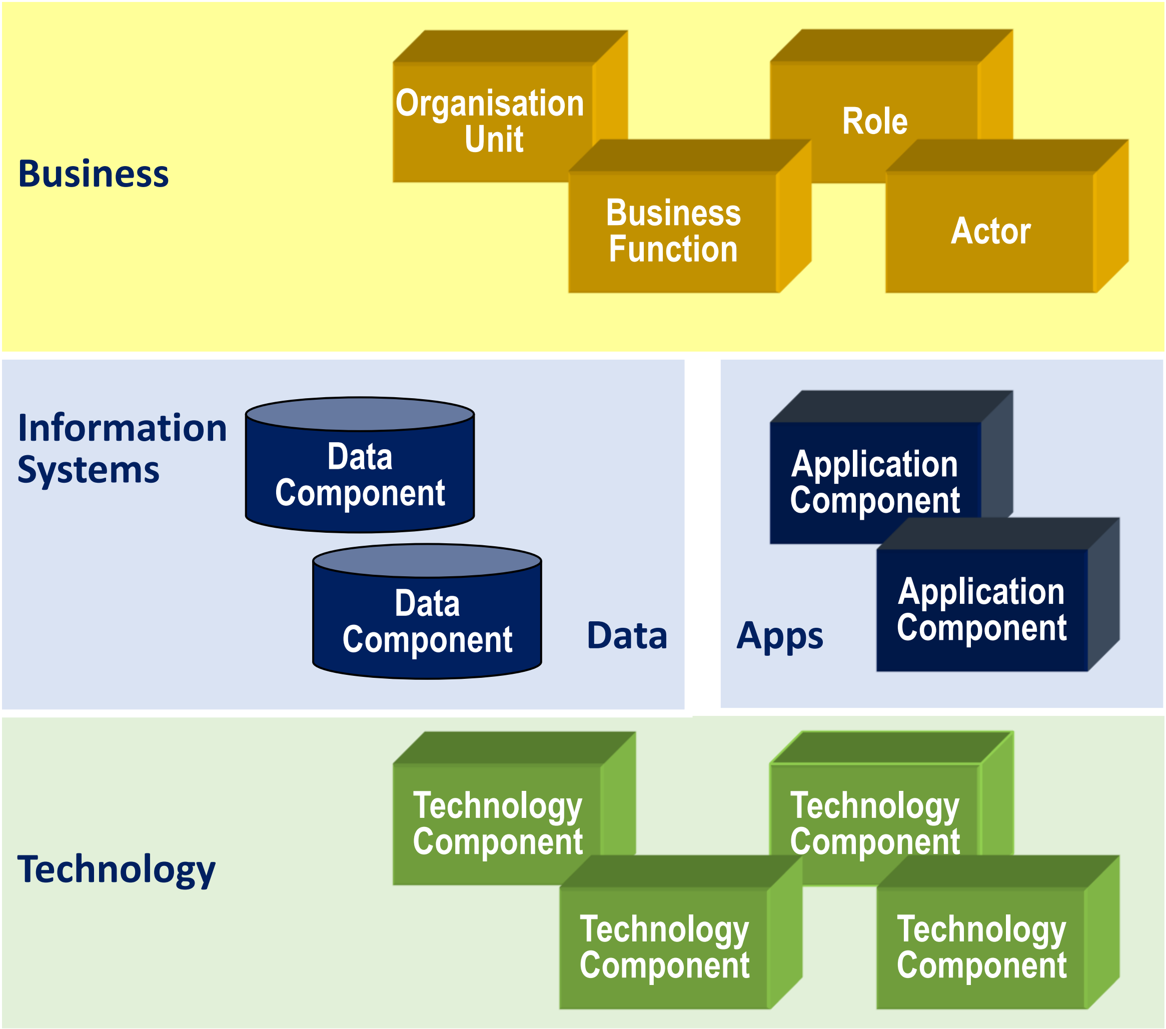
In ArchiMate’s model of business systems

- Externally visible *services* and *interfaces* encapsulate internal
  - Behavioral elements - called *processes* and *functions.*
  - Active structures - *actors* who play *roles*

	Behaviors	Structures	
External view	Business Service	Business Interface	
Internal view	Process Function	Role	Logical
		Actor	Physical

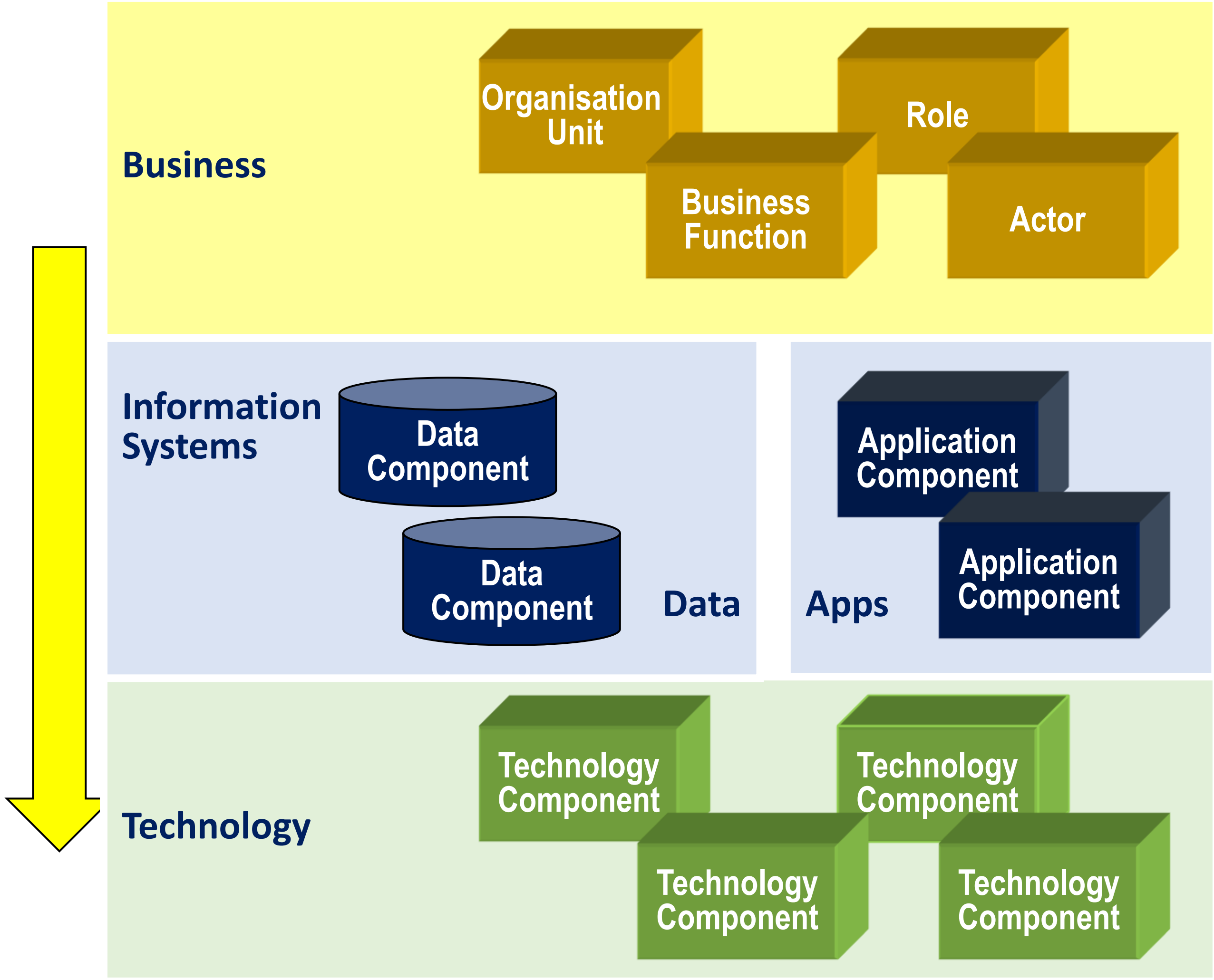
In TOGAF's meta model of an EA

- The structural components of business systems are arranged in four architecture domains



Principle 1: Business before technologies

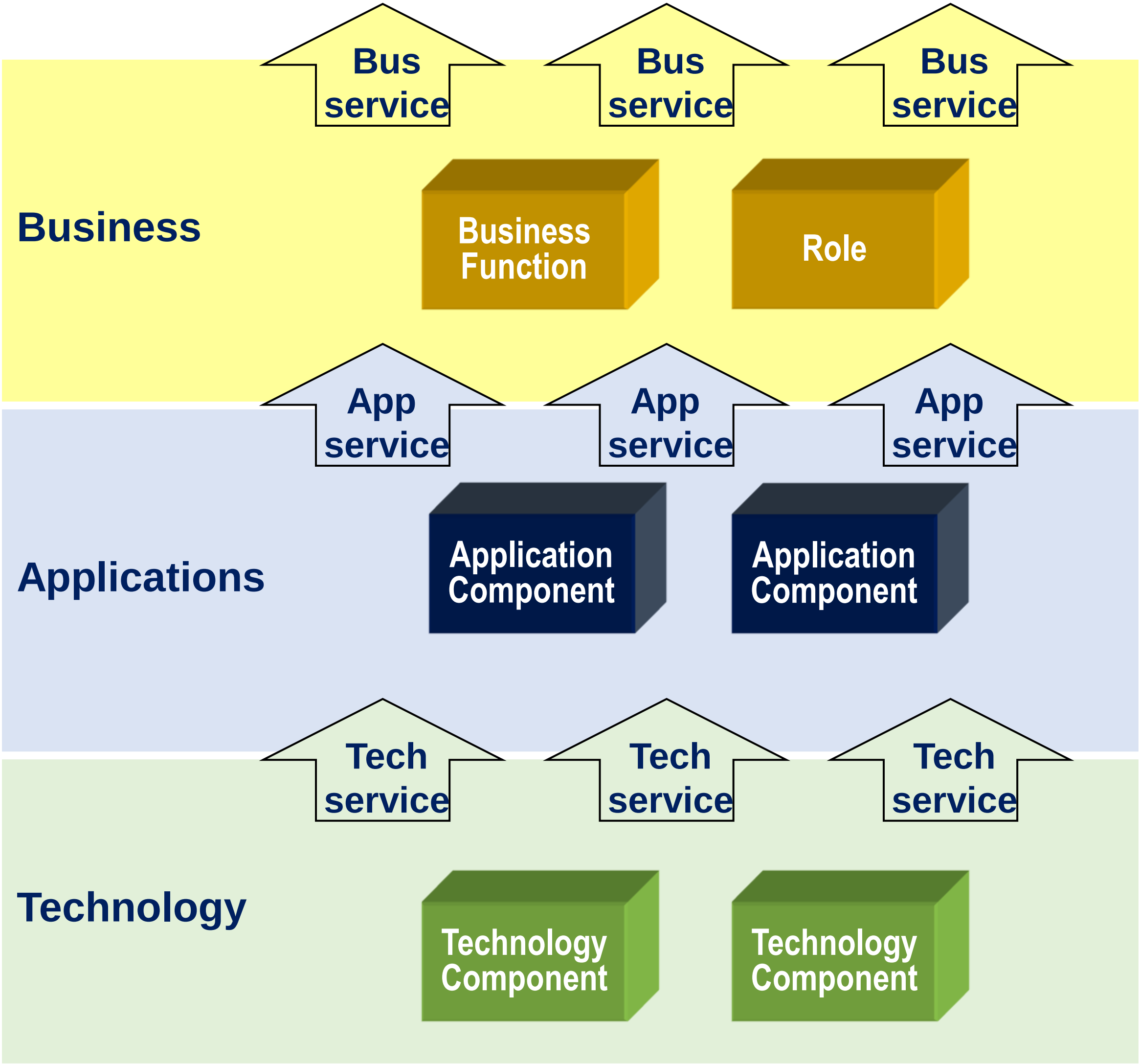
- The EA principle is to consider first what must be enabled.
- A business procures technologies to enable its business functions



An extract from training at http://avancier.website

Service-oriented architecture

- Modern service-oriented EA frameworks see architecture domains as layers in a client-server hierarchy
  - business services
  - information/app services and
  - technology services.



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)

Structure and behavior: which comes first?

- In system *operation*, structures perform behaviors
  - Actors connect in a structure to perform required behaviors
- In system *design*, behaviors determine structures
  - First define the required behaviors
  - Then define the structures needed to perform the behaviors

Principle 3: Behaviors before structures

- Structures are built, hired or bought to perform or enable required behaviors

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

- A service may triggered via an interface, or 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 ↓

Mapping business architecture terms to generic ones

- Value Streams are Processes
- Functions and Capabilities are Logical Business Components
- Organizations and Actors are 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 business system concepts, 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**

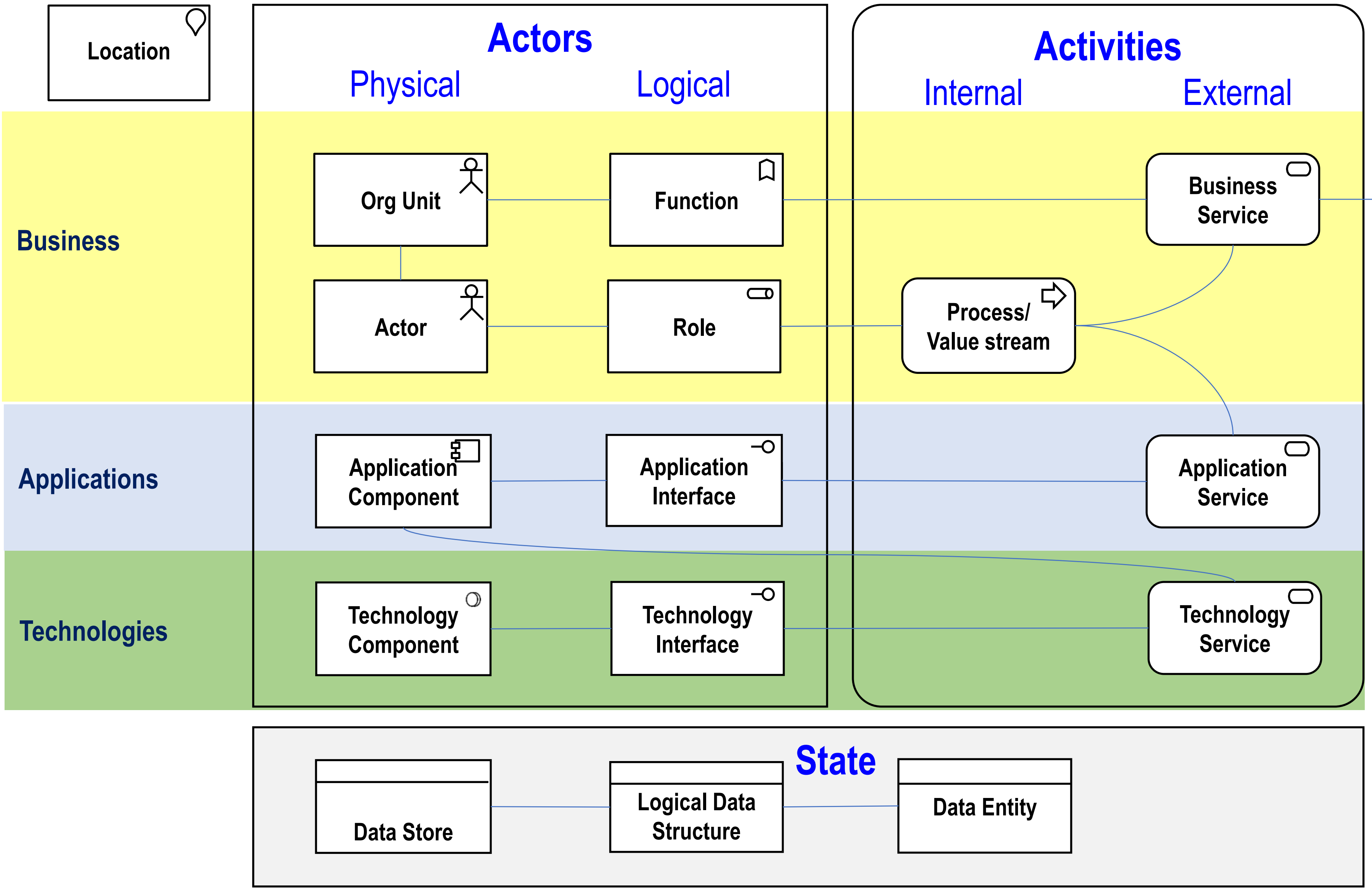
A classification compatible with ArchiMate and TOGAF

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	Business Interface Function/ Capability Role	*Logical*
	Process/ Value Stream	Organization Unit Actor	*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	*Logical*
		Application Component	*Physical*
Technology architecture Services that Technologies provide (as catalogued in TOGAFs TRM) are obscure in most modern EA practice	Technology Service	Technology Interface	*Logical*
		Technology Component	*Physical*

Actors perform activities to maintain system state and meet aims



Bus. Services can be coarse-grained. E.g. Sell a house.

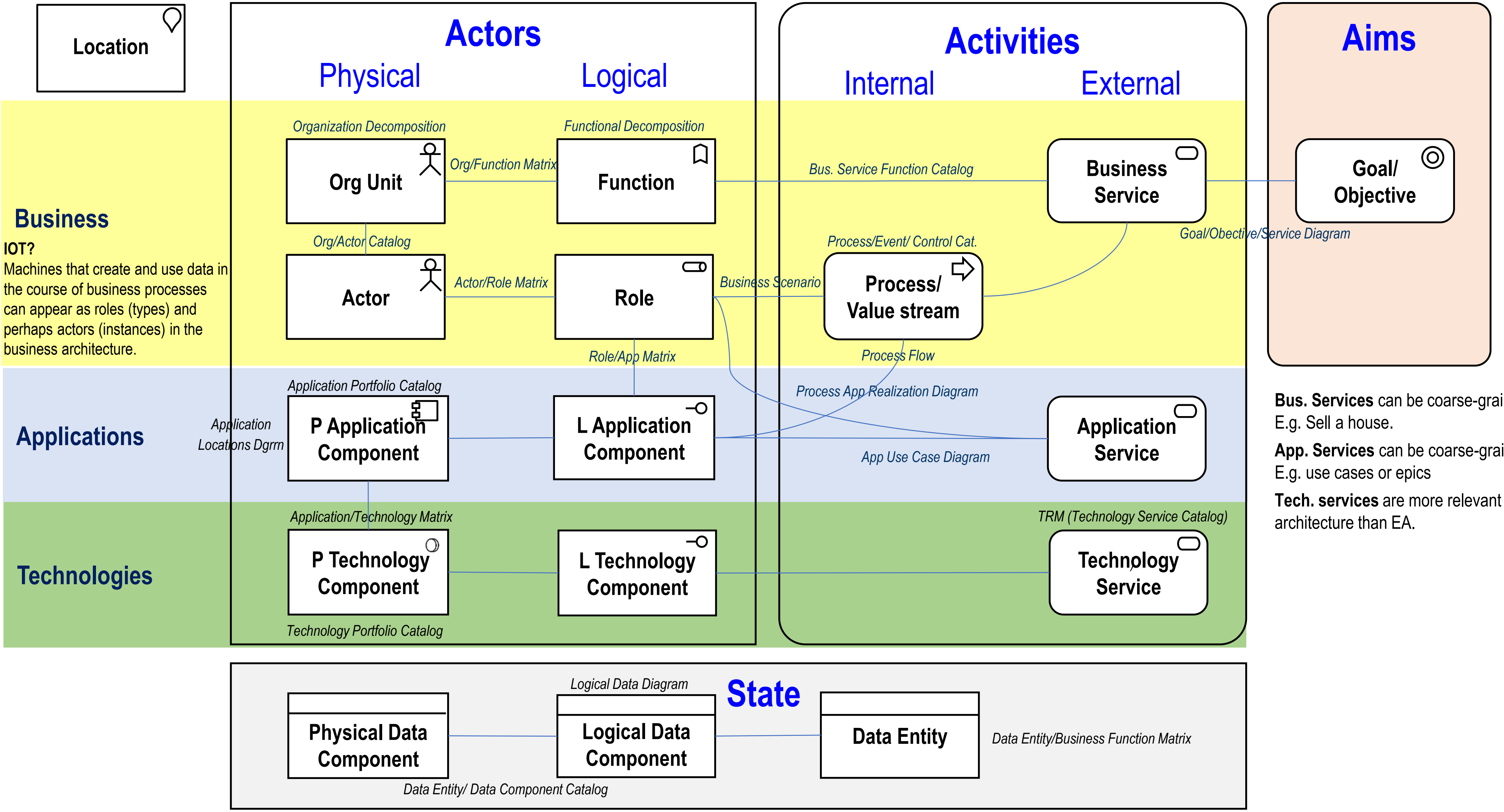
App. Services can be coarse-grained. E.g. use cases or epics

Tech. services are more relevant in IT architecture than EA.

An extract from training at http://avancier.website

Recording *all* this in an EA repository is impractical. You document what you find useful.

A meta model connecting TOGAF 9 artefacts



Location

Actors

Physical

Logical

Activities

Internal

External

Aims

Business

IOT?
Machines that create and use data in the course of business processes can appear as roles (types) and perhaps actors (instances) in the business architecture.

Organization Decomposition

Org Unit

Org/Function Matrix

Functional Decomposition

Function

Bus. Service Function Catalog

Business Service

Goal/ Objective

Goal/Obective/Service Diagram

Org/Actor Catalog

Actor

Actor/Role Matrix

Role

Business Scenario

Process/Event/ Control Cat.

Process/ Value stream

Role/App Matrix

Process Flow

Process App Realization Diagram

Applications

Application Portfolio Catalog

Application Locations Dgrm

P Application Component

L Application Component

Application Service

App Use Case Diagram

Technologies

Application/Technology Matrix

P Technology Component

L Technology Component

TRM (Technology Service Catalog)

Technology Service

Technology Portfolio Catalog

State

Logical Data Diagram

Physical Data Component

Logical Data Component

Data Entity

Data Entity/Business Function Matrix

Data Entity/ Data Component Catalog

Bus. Services can be coarse-grained. E.g. Sell a house.

App. Services can be coarse-grained. E.g. use cases or epics

Tech. services are more relevant in IT architecture than EA.

An extract from training at http://avancier.website

Which are true?

1. Services are defined as internal system elements
2. A component is a package of functionality or capability that offers one or more services
3. Service encapsulate processes
4. Components perform processes
5. Ideally, the same service should not be provided by different components
6. A component can be seen as a subsystem
7. A component is elementary; it cannot be decomposed
8. Components can be classified on scales
    - from Business to Technical and
    - from Logical to Physical

POSTSCRIPT: ten troublesome terms

- Ludwig von Bertalanffy wanted general system theory to unify all sciences.

- Yet today, system terms
  - holistic, emergence, behavior, non-linear, fractal,
  - complexity, complex adaptive system, self-organization, chaos

- are used so differently that no useful generalization is possible.
- or so loosely, the meaning is unclear

Thinking holistically?

1. Considering how *interactions* between particular selected “parts” produce results/effects
    - rather than considering those parts in isolation

2. NOT considering the “whole” of a physical entity
    - Since every system of interest an abstraction from a reality
    - Every set of “parts” is a selection from infinite conceivable parts
    - It is a reduction of the whole, made with some interest in mind
    - Which means infinite systems may be abstracted from one enterprise

Emergent properties?

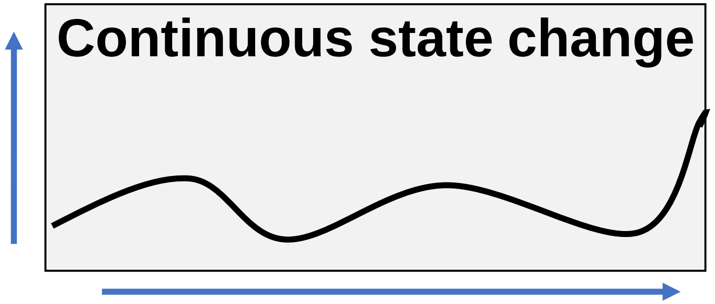
1. Results/effects that emerge from interactions between “parts”
   - E.g these properties emerge from autonomous actors following simple rules.
     - the V shape of flight of geese
     - the shimmering of a shoal of fish
     - the price of fish - set by customers and suppliers balancing their needs

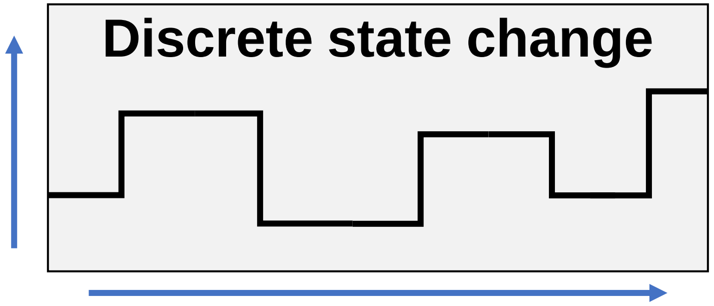
2. NOT surprising or unexpected outcomes
   - True, emergent properties may seem mysterious in nature
   - But every designed system is designed to produce emergent properties

Behavior? Ambiguous!

1. A process (usual meaning in EA)
    - any action, activity, operation or procedure that takes time to perform

2. A state change trajectory (usual meaning in “system dynamics”)
    - a “line of behavior” over successive state changes

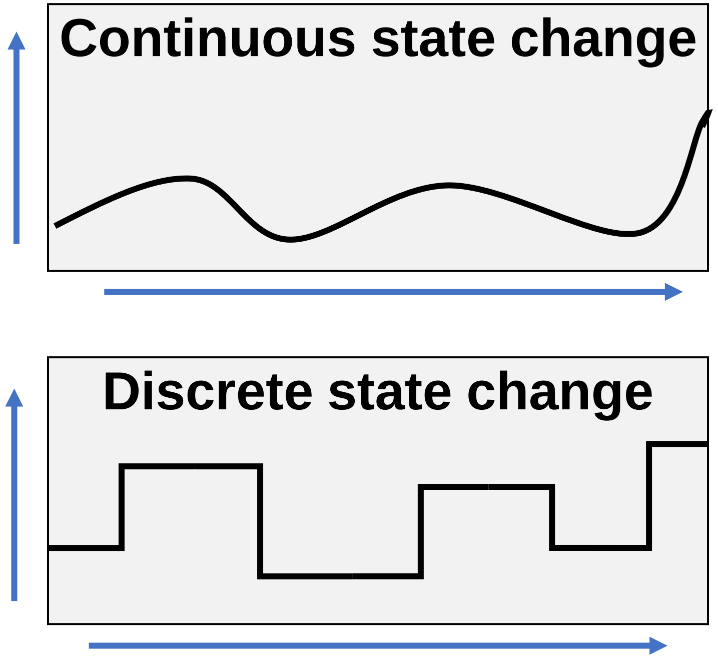

Continuous state change


Discrete state change

Non-linear?

1. Means not in a straight line
   - Curved or jagged
     - As state variable values may change.
       - unit prices in a stock market, or
       - the number of people infected by a virus
   - A homeostatic sine wave
   - An exponential increase or decrease

2. NOT human rather than mechanical!



Fractal?

1. A recursive hierarchy
   - Zoom in or out, you find the same structure

2. NOT organization hierarchy or functional decomposition
   - Zoom in, you divide one element into different elements
   - Zoom out, you group different elements into one

Complexity? Unclear!

- Each dimension of a system might be measured in terms of complexity
  - A structure in which actors, activities or aims are connected
  - The logic of rules or processes that control activities
  - The structures of I/O material or information flows
  - The structures of memories that actors maintain
  - The trajectories of state changes over time.

- Scores of complexity measures have been proposed.
- Measurement is so difficult people make subjective comparisons.

Complex Adaptive System? Unclear!

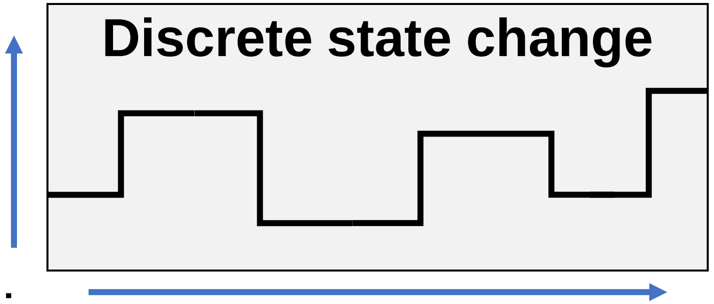
- A CAS is often simple by any normal complexity measure!

- The term CAS usually implies
  - “self-organizing” behavior
  - non-linear state change

More here:

http://grahamberrisford.com/AM%204%20System%20theory/What%20is%20a%20complex%20system.htm

Complex **Adaptive** System? Ambiguous!



1. Adaptation by state change
   - A system’s state advances in response to inputs
   - It may thus “adapt” to its environment - as in homeostasis
   - In EA, systems create and update data stores

2. Adaptation by system mutation
   - A system’s character may change from one generation to another
   - As it does in biology and in software engineering
   - In EA, system mutation implies change control

Complex Adaptive **System?** Ambiguous!

1. Activity system thinking
   - About a set of inter-related activities performed by actors

2. Social “system” thinking
   - About a set of inter-related actors who act in ad hoc or innovative ways

- EA addresses regular deterministic business system activities
- These are complex and challenging enough!
- Much important business activity is ad hoc, psychological and sociological.
- Let us not pretend our architects are employed to design and govern those activities as well!

Self-organization? Ambiguous!

- Emergence alone? Surely not?
- “Goal-seeking" state change?
  - A entity is drawn to an "attractor" state and resists being moved from that state.
  - As in homeostatic biological and electro-mechanical control systems.
- Self-assembly?
  - An entity grows incrementally by adding more elements or actors to its body.
  - E.g. the growth of a crystal in a liquid, or a plague of locusts.
- Self-improvement – by system mutation? as in social entity thinking?
  - Ashby and Maturana rejected this as undermining the concept of a system.
  - I reconcile them by introducing the concept of a meta system.
  - One person can play a role in both system as actor and meta system as system definer.

Chaos? Ambiguous!

Actors: two pendulums; one attached to the bottom of the other.
Activities: swing governed by Newton’s laws of motion.
State variables: arm height, arm length, ball weight.
Small changes in the initial state lead to wildly different outcomes.
https://twitter.com/i/status/1213398926696632321

1. In activity system theory (as in maths) chaos means
   - though activities are regulated by deterministic rules
   - the system changes state in apparently random or irregular ways
   - and is highly sensitive to initial conditions

2. In social entity thinking, chaos might mean
   - actors’ activities are not rule-bound and/or
   - activities are disorganised or conflicting and/or
   - no pattern can be detected how actor’s interact

More, edited from Will Harwood

- Catastrophe and chaos are different ideas.
  - A system caught in a strange attractor is chaotic.
  - Yet undergoes no catastrophe in a technical/catastrophe theory or everyday sense.

- Predictable, controllable and stable are different ideas.
  - A system may be predicable without being controllable
  - A system may be both the above yet not stable in the accepted technical sense.

- "The edge of chaos" is problematic idea.
  - Informally, a phase transition in a system from a predictable regime to chaotic regime.
  - In system science, both predictable and chaotic regimes are deterministic.
  - The term has no precise and general definition across domains where the term is used.
  - The phrase gives the impression of something profound when little has been said.

Ashby’s “Introduction to Cybernetics” 1956

- On dynamics
  - continuous dynamics can be simulated using discrete dynamics
- On abstraction
  - many physical entities can realize one activity system
  - many activity systems can be realized by one physical entity
- On regularity
  - an activity system applies a set of rules to a set of state variables.
- On adaptation
  - system state change differs from system mutuation/rule change.
- On self-organization
  - self-assembly differs from self-improvement.
- The law of requisite variety
  - controllers must recognize if not remember the variety they seek to control.

Further reading

A long introduction systems thinkers and their ideas https://bit.ly/2yXGImr

Two articles on basic ideas

- Distinguishing activity systems from entities https://bit.ly/2w5XKNK
- Ashby’s core ideas about activity systems https://lnkd.in/eCRp3H4.

Two articles with novel ideas about systems

- “Third order cybernetics” https://bit.ly/2UR6rql
- A philosophy of systems https://lnkd.in/dQNhNbd

Much more on my system theory page at https://bit.ly/2yXGImr