Business Capabilities as Formalised Social Systems

By Graham Berrisford
What are the essential elements of a society?

“The sociological tradition suggests two alternatives: either [actors] or activities.” (David Seidl, 2001)

**Actor-centric view**

Given some actors, what can they do? An **actor** is a discrete structural entity able to perform activities.

**Activity-centric view**

Given some activities, what actors are needed? An **activity** is an event-triggered element of behaviour performable by actors.

EA is about **activity systems**
Social systems

Social Group (actor-centric view)

Is a set of **physical actors** who communicate as they choose with each other.

Social System (activity-centric view)

Is defined by its **logical roles** (as in a choir or business system).

An assignment to a role is only a small part of a person (Boulding, 1956)

**EA assigns activities to logical roles and functions**
## Individual level: Roles and Actors

- **Role:** The usual or expected *function* of an actor.” TOGAF
- A role describes a group of activity types that one actor can perform

<table>
<thead>
<tr>
<th>People</th>
<th>Behaviour</th>
<th>Structure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Activity Types</td>
<td>Roles (functions)</td>
</tr>
<tr>
<td>Reality</td>
<td>Activities (tasks)</td>
<td>Actors (assignments)</td>
</tr>
</tbody>
</table>

“A business actor is an organizational entity capable of performing behavior.” ArchiMate

- **Actor:** A person, organization, or system that has a role.”
  - “An actor may have a number of roles.”
- Actors (not roles) perform activities.
- Actors are sometimes named in an architecture description.
“**Function:** describes units of business capability at all levels of granularity” TOGAF

A function describes a group of processes that org units can perform.

<table>
<thead>
<tr>
<th>Organisat’n</th>
<th>Behaviour</th>
<th>Structure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Process Types</td>
<td>Functions</td>
</tr>
<tr>
<td>Reality</td>
<td>Processes</td>
<td>Org Units</td>
</tr>
</tbody>
</table>

“Structured Analysis maps functions onto organizational units.”

“**Org unit:** A self-contained unit of resources with goals, objectives, and measures.”

Org units (not functions) perform processes.

Org units are often named in an architecture description.

“A business actor is an organizational entity capable of performing behavior.” ArchiMate
**Social system design**

<table>
<thead>
<tr>
<th>Social system</th>
<th>Behaviour</th>
<th>Structure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Activity Types</td>
<td>Roles</td>
</tr>
<tr>
<td>Reality</td>
<td>Activities</td>
<td>Actors in Roles</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Behaviour</th>
<th>Structure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logical Description</td>
<td>Activity Types</td>
<td>Roles</td>
</tr>
<tr>
<td>Physical Description</td>
<td>Actor Specifications</td>
<td>Roles</td>
</tr>
<tr>
<td>Deployment</td>
<td>Performances</td>
<td>Actors in Roles</td>
</tr>
</tbody>
</table>

**TOGAF and ArchiMate**

<table>
<thead>
<tr>
<th>ArchiMate</th>
<th>Behaviour</th>
<th>Structure</th>
</tr>
</thead>
<tbody>
<tr>
<td>External</td>
<td>Services</td>
<td>Interfaces</td>
</tr>
<tr>
<td>Internal</td>
<td>Processes</td>
<td>Components</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>TOGAF</th>
<th>Behaviour</th>
<th>Structure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requirement &amp; Context</td>
<td>Needs</td>
<td>Requirements</td>
</tr>
<tr>
<td>Architecture Continuum</td>
<td>Services and Processes</td>
<td>Logical Components</td>
</tr>
<tr>
<td>Solutions Continuum</td>
<td>Physical Components</td>
<td>Deployment Solutions</td>
</tr>
<tr>
<td>Deployed Solutions</td>
<td>Deployed Components</td>
<td></td>
</tr>
</tbody>
</table>

“EA regards the enterprise as a system, or system of systems.” TOGAF
As business systems have formalised social systems
Further…. software systems have formalised business systems

- **A role** describes a group of activity types that one actor can perform.

- **A class** describes a group of operations that one object can perform.

<table>
<thead>
<tr>
<th>People</th>
<th>Behaviour</th>
<th>Structure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Activity Types</td>
<td>Roles</td>
</tr>
<tr>
<td>Reality</td>
<td>Activities</td>
<td>Actors (assignments)</td>
</tr>
</tbody>
</table>

- Actors (not roles) perform activities.
- Actors are sometimes named in a human system description.

<table>
<thead>
<tr>
<th>Software</th>
<th>Behaviour</th>
<th>Structure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Operations</td>
<td>Classes</td>
</tr>
<tr>
<td>Reality</td>
<td>Procedure Executions</td>
<td>Objects</td>
</tr>
</tbody>
</table>

- Objects (not classes) perform operations
- Objects are rarely named in a software specification.
The importance of information to society and Business Architecture

By Graham Berrisford

Hundreds of thousands of texts in the Sumerian language have survived, inc. business letters, receipts, glossaries, laws, daily records...

Tablet recording the allocation of beer: 3100–3000 BC

by BabelStone - Own work. Licensed under CC BY-SA 3.0 via Commons - https://commons.wikimedia.org/wiki/File:Early_writing_tablet_recording_the_allocation_of_beer.jpg#/media/File:Early_writing_tablet_recording_the_allocation_of_beer.jpg
What kind of business capabilities is EA concerned with?

► Business systems can be seen as formalised social systems
  ■ in which actors communicate, play given roles and follow given rules.

► Business transactions can be seen as formalised social transactions
  ■ in which facts are communicated using standard messages and shared memory spaces.

► The aim is to enable, support, monitor or direct business behaviour (activities, services…).
  ■ Structure - the focus of building architects - is secondary.
► The behaviour is usually orderly, repeatable and under change control (not changed without approval).
What kind of business capabilities is EA concerned with?

- “EA is the determinant of survival in the **Information Age**.” (Zachman)
- “**information-intensive** organisations…is the main focus” (ArchiMate v2.1)
- “Today’s CEOs know that the effective management and exploitation of **information** through IT is a key factor to business success.” (TOGAF 9.1)
- “EA structures and gives context to activities delivering concrete business outcomes, primarily but not exclusively in the **IT [IS] domain**.” (TOGAF 9.1)
Data v Information in the NIST Enterprise Architecture Model

- A reference model promoted in U.S. federal gov.

- **Information** is **data** at the point of creation or use by an actor
  
  (See “Information theory” on the “Sense and nonsense in system theory” page at [http://avancier.website](http://avancier.website).)

- In practice, the terms are interchangeable. Why use a long word when a short one will do?
“Business change” working alongside EA

**Actor-centric view**

Human aspects: groups and customs of actors may be addressed using a socio-cultural approach by a business change team.

**Activity-centric view**

EA focuses on activities that create or use data that represents business facts.

*EA doesn’t and can’t describe every activity that happens in a business.*
Design sequences
in Enterprise and Solution Architecture

By Graham Berrisford
Premises on which activity system modelling is based

- **EA models**
  - activity systems that are composed of
  - actors (discrete structural elements) that communicate and cooperate in
  - event-driven activities (discrete behavioural elements) and remember
  - the state of actors and activities.
If structured analysis is completed then

Activities shown as Elementary Business Processes (EBPs)

Hairdressing: Value Stream / Scenario / Process

- Answer telephone call
- Make appointment
- Shampoo, condition and rinse hair
- Talk to client about requirements
- Cut hair with clippers, scissors
- Dry hair and style
- Operate cash register
- Maintain client record

Appear at the bottom of a structural view as Elementary Business Functions (EBFs)

Hairdressing: Function Hierarchy / Capability Map

Client relationship management
- Talk to clients about requirements
- Advise clients on hair care
- Maintain client records

Core business
- Shampoo, condition and rinse hair
- Cut hair with clippers, scissors
- Dry hair and style
- Shave and trim facial hair

Reception and payment
- Answer telephone calls
- Make appointments
- Operate cash register

Copyright Avancier Ltd 2008 to 2015
Assigning behaviour to structure in one diagram

Swim lanes show structure
- Actor
- Role or Function

Arrows show behaviour
- Event
- Trigger
- Activity

Copyight Avancier Limited 2008 to 2015
The starting point

- Encapsulate the actors and activities (the internal view)
- Start from the events/requests actors must respond to (the external view)

**After general system theory**

- Events & Responses
- Boundary
- Activities
- Actors in Roles

**A generic architecture meta model**

- Event/Service
- Interface
- Process
- Component

Actors playing roles in a system respond to input events by performing activities appropriate to their roles.

Components (which present services via interfaces) respond to service requests by performing processes.
The same principle applies at every level

► You always start with the responses you want from the system
Four design sequences in architecture frameworks

- **Logical to physical**
  - abstract specification before concrete implementation

- **External to internal**
  - required services/products before internal processes and roles/actors/components

- **Behaviour to structure**
  - services and process sequences before roles/actors/components.

- **Business to technology**
  - human actors before computer actors.

- There are reasons to deviate from these sequences in practice, and to “reverse engineer”

- But methods for governing design and implementation against a specification usually start with these presumptions
A general process for formalisation of capabilities

- **Design the target**
  - Define required services
  - If need be, design processes to deliver the required services.
  - Assign required services and/or process steps to logical interfaces, functions or roles.
  - Change, hire, buy or build physical actors/components to realise logical Interfaces and perform process steps.
A general process for rationalisation of a messy system estate

► Analyse the baseline
  ■ catalogue baseline Components (org. units, functions, roles or actors).
  ■ Abstract the discrete Services provided by those baseline Components.
  ■ De-duplicate baseline-provided Services

► Design the target
  ■ Define required services
  ■ If need be, design processes to deliver the required services.
  ■ Assign required services and/or process steps to logical interfaces, functions or roles.
  ■ Change, hire, buy or build physical actors/components to realise logical Interfaces and perform process steps.
A general process for designing & changing a business capability

Motivation
1. Gather drivers, goals, principles etc.
2. Identify stakeholders, concerns etc.

Business architecture
1. Define new/changed business services required.
2. Define logical business processes and/or functions to provide required business services.
3. Define business roles to perform elementary business process steps and/or business functions.
4. Change, hire, buy or build organisation units and human actors
   (revise business organisation structure as need be).

Information systems architecture
1. Define new/changed information system services (use cases) required.
2. Define I/O data flows (in service contracts or use case headers) and data stores.
3. If need be, define logical application processes (as in use case paths)
4. Change, hire, buy or build physical applications to support and enable use cases
   (revise business application portfolio as need be).

Infrastructure Technology architecture
1. Define new/changed platform services required (revise the TRM as need be).
2. Don’t define logical platform processes, since they are given by vendors behind APIs.
3. Hire or buy physical technologies to provide required APIs
   (revise technology component portfolio as need be).
How AM harmonises TOGAF and ArchiMate for Enterprise and Solution Architects

By Graham Berrisford
The ambition of EA wrt formalised social systems

Business Architecture (Description)

Event Types

"An EA" is the enterprise’s architecture description, recorded some kind of architecture repository.” TOGAF

Activity Types

Roles

Standardise, integrate and share common services between business operations

Realise architecture descriptions in operational systems

Review compliance of business operations to architecture

Business Operations (Reality)

Physical actors acting (supposedly) according to system descriptions.

Events Responses

Activities

Actors in Roles

Copyright Avancier Limited 2008 to 2015
Things essential to EA

- Formalisation of information in messages and memories
- Formalisation of roles and rules
- Formalisation of change control
- Abstraction of system descriptions from operational systems
What does formalisation mean?

► EA frameworks are based on two presumptions few make explicit.

► As UML 2.1 puts
1. “all behavior in a modeled system is ultimately caused by actions executed by so-called “active objects”
2. “behavioral semantics only deal with event-driven, or discrete, behaviors”
Formalisation typically follows this sequence:

**Logical Description**
Define events of interest
Define the required activities
Group into logical roles

**Physical Reality**
Hire, buy or build physical actors to act in logical roles.

Be cautious about naming actors
Architects abstract from the infinite complexity of real-world structures and behaviours.

System descriptions

“An EA” is the enterprise’s architecture description.” TOGAF

Architects

Operational systems

Requirements and constraints

create and use

observe and envisage

idealise
Aligning the generic meta models of different standards

This reference model aligns TOGAF and ArchiMate standards

Our generic meta model

- Event/Service
- Interface
- Process
- Component
- Behaviour to Structure

TOGAF
- Service
- Logical Component
- Process
- Physical Component
- Behaviour to Structure

ArchiMate
- Service
- Interface
- Behaviour Element
- Structure Element
- Behaviour to Structure

Avancier Methods

► are useful with all architecture frameworks that share similar domains and entities

► http://avancier.website

Avancier Methods

BCS E&SA reference model

TOGAF
The Open Group

ArchiMate
Language Framework

CSC’s domains of change (POLDAT)

IBM’s view EA

EA as Strategy”
MIT