Only direct relations are shown. Abstract elements are expanded so only real metamodel elements are shown. Uses the Mastering ArchiMate 9-colour scheme to assist learning and quick reading.
### Motivation Aspect

- **(Driver)**
- **(Goal)**
- **(Outcome)**
- **(Principle)**
- **(Requirement)**
- **(Constraint)**
- **(Stakeholder)**
- **(Assessment)**
- **(Value)**

**Meaning**

Any structural or behavioural element from Strategy or Core domains, any Composite element.

**NOTE:** explicit in the metamodel and not shown here:

- Stakeholder can be Associated with any other element
- Any Motivation element, except Stakeholder, may Influence any other Motivation element

### Strategy Layer

- **(Grouping)**
- **(Resource)**
- **(Course of Action)**
- **(Capability) or (Value Stream)**

**Core Structural Element (e.g. Actor, Role, Application Component, Node, Interface, Artifact, Material, Object)**

**Core Behavioural Element (e.g. Process, Function, Service)**

### Composites

- **(Location)**
- **(Grouping)**
- **(Capability) or (Value Stream)**

**Can Aggregate any Element or Relation**

### Implementation & Migration Layer

- **(Grouping)**
- **(Location)**
- **(Implementation Event)**
- **(Plateau)**
- **(Gap)**
- **(Work Package)**
- **(Deliverable)**

**Any behavioural or structural element from the Core or Strategy Domains.**

**Any element from the Motivation Aspect except Stakeholder**

(Realisation only, though Influence can be used (derived))

---

This is the very fine print. Document locked against editing due to external licensing demands.
### Business Layer Elements

- **(Business Actor)**
  - Represents actual people or groups of people, such as departments, companies, etc., e.g., 'Audit Department'.

- **(Business Role)**
  - Represents a virtual 'actor' (defined as responsibility for behaviour), e.g., 'Manager'.

- **(Business Collaboration)**
  - Represents a collaboration of actors/roles for behaviour that cannot take place if the members are not all there, e.g., 'dance couple'.

- **(Business Interface)**
  - Represents how an actor/role/collaboration can be accessed, e.g., 'Public phone line'.

- **(Business Function)**
  - Behaviour at the business level grouped on the basis of common characteristics (inside-out). E.g., 'Finance', 'Cleaning'.

- **(Business Process)**
  - Behaviour at the business level grouped on the basis of outcome (outside-in). E.g., 'Create invoice', 'Open shop'.

- **(Business Interaction)**
  - Represents behaviour that is usable/visible from the 'outside', performed by an interface, e.g., 'Helpdesk support', 'Flower sales'.

- **(Business Service)**
  - Represents a 'state change' modelled at the business level, e.g., 'Customer enters', '9 o'clock', or 'Start of meeting'.

- **(Business Event)**
  - Represents a 'state change' modelled at the business level, e.g., 'Launch nuke' (one hopes).

- **(Business Object)**
  - Represents at the business level something passive that is 'acted upon', e.g., 'Bill', 'Investment', 'Claim'.

- **(Contract)**
  - Specialization of Business Object to represent a contract.

### Technology Layer Elements

- **(Node)**
  - Generic technology element. Often used to aggregate other technology elements, e.g., a server as Node that contains Device and System Software.

- **(Device)**
  - Represents actual computing hardware, e.g., 'x86 blade'.

- **(System Software)**
  - Software system at the technology level. Generally used for platforms, e.g., 'PostgreSQL', 'Windows 2016 Server', 'Java SE'.

- **(Technology Interface)**
  - How an active technology element can be accessed, e.g., 'Port 25', 'SMTP Protocol'.

- **(Technology Collaboration)**
  - Represents a combination of technology elements that only together can perform certain behaviour.

- **(Path)**
  - Logical representation of a distribution channel between Nodes, etc., e.g., 'data replication', 'finished product exchange'.

- **(Communication Network)**
  - Data transport facility, e.g., 'datacenter network'.

- **(Facility)**
  - Higher level grouping of physical actors, e.g., 'Brick factory'.

- **(Equipment)**
  - Machinery that can perform physical behaviour, e.g., 'Steel mill'.

- **(Artifact)**
  - Raw data, e.g., 'database files', 'software distribution', 'PDF file'.

- **(Material)**
  - Physical stuff, e.g., 'Wood blocks', 'Nails'.

- **(Distribution Network)**
  - Logical representation of a distribution channel between Nodes, etc., e.g., 'data replication', 'finished product exchange'.

**NOTE:** The element descriptions on this page are not the official ones from The Open Group and may be ‘opinionated’.
Overview of ArchiMate® 3.1 Elements (2/2)

Diagrams by Gerben Wierda.

Application Layer Elements
- (Application Component)
- (Application Interface)
- (Application Collaboration)
- (Data Object)

Strategy Layer Elements
- (Capability)
- (Resource)

Motivation Aspect Elements
- (Goal)
- (Outcome)
- (Driver)
- (Assessment)
- (Value)

Implementation & Migration Layer Elements
- (Work Package)
- (Implementation Event)
- (Deliverable)

NOTE: The element descriptions on this page are not the official ones from The Open Group and may be ‘opinionated’.

Copyright © 2020 Gerben Wierda. It can be freely used, copied, printed, etc. Version: 20200522. ArchiMate® is a registered trademark of The Open Group. The ArchiMate 3.1 standard is © The Open Group.