4.1 INTRODUCTION

The previous chapter (Chapter 3) acknowledged current theories that are evident in various alignment approaches, e.g. systems theory, systems engineering and the basic systems design process, different paradigmatic schools of thought, and the ISO/IEC/IEEE 42010 standard on architecture description. The chapter also described six alignment approaches and identified the need to compare various alignment approaches with one another in terms of business-IT alignment (see section 3.1). Chapter 3 concluded with eight *other alignment approaches,* which are also referenced in this thesis.

Schekkerman (2004) aptly describes the explosion of enterprise architecture frameworks with the title of his book 'How to survive in the jungle of enterprise architecture frameworks'. The number of relevant EA frameworks emphasises the need to provide a common reference model in order to discuss and compare various alignment approaches with one another. The purpose of this chapter is to recognize the knowledge embedded in current alignment approaches by inductively creating a model that will highlight prominent themes/patterns evident in each of these alignment approaches. This chapter answers the second research question, namely:

What model is required to contextualise different business-IT alignment approaches?

This chapter applies the theory of Chapter 3 through an inductive development process to develop the BIAM (Business-IT Alignment Model)[1]. The chapter starts with the inductive development process that was followed, emphasising the contributions of the six alignment approaches previously discussed in section 3.3, since each approach differs in business-IT alignment intent, scope and alignment means. Section 4.2 repeats and extends the research design (exploratory design, previously discussed in section 2.6.3), whereas section 4.3 details the components of the proposed BIAM.

4.2 THE BIAM CONSTRUCTION PROCESS

This study used inductive reasoning (see Figure 44), discussed previously in section 2.6.3, to derive a Business-IT Alignment Model (BIAM).

[1] The content of Chapter 4 is based on: De Vries, M. (2010). A framework for understanding and comparing enterprise architecture models. *Management Dynamics, 19*(2), 17-29.

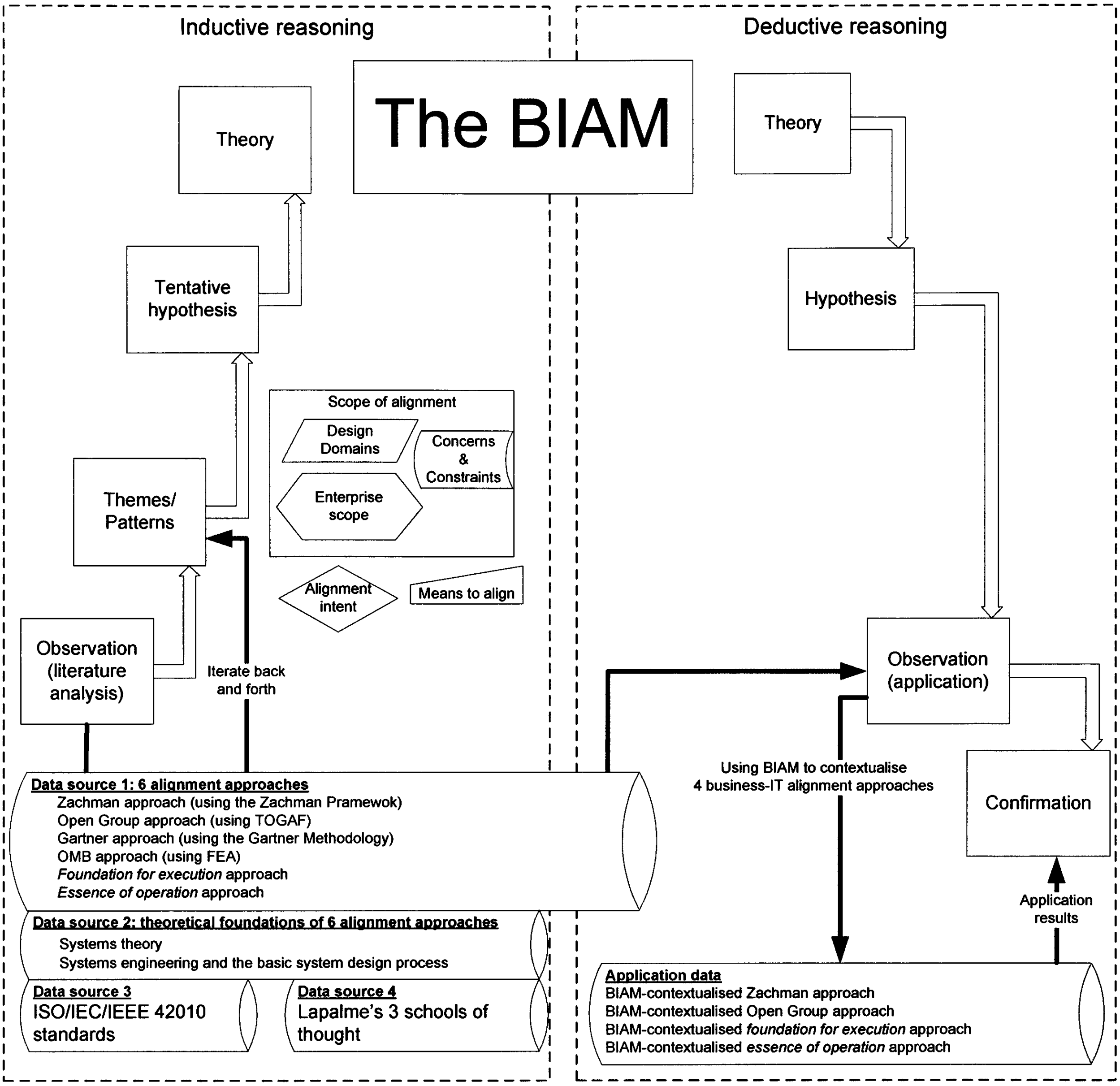


Figure 44: Exploratory design for building and applying the BIAM (duplicate of Figure 16)

As stated in section 2.6.3, an exploratory research design was used as a supplementary component of a mixed methods design, to develop the BIAM. Furthermore, a literature review was used as the data-gathering method, *inductively* formulating the main components of the BIAM. Subsequently the conceptual BIAM is applied (in Chapter 5) in a *deductive* way to demonstrate the interpretation and use of the model in terms of four theoretical alignment models.

As mentioned in section 2.6.3, this study used four *main data sources* in constructing the BIAM:

1. Six current alignment approaches (discussed in section 3.3).
2. Theoretical foundations of the six alignment approaches, which include systems theory (discussed in section 3.2.1), systems engineering and the basic system design process (discussed in section 3.2.2).
3. The ISO/IEC/IEEE 42010 standards (discussed in section 3.2.4).
4. Lapalme's three schools of thought (discussed in section 3.2.3).

A *secondary data source* (eight *other alignment approaches*, discussed in section 3.4) was used to provide additional motivation and explanation for the BIAM components.

In this thesis, the initial development of BIAM (called BIAF (De Vries, 2010)), was extended to acknowledge the three different schools of thought on alignment approaches, as defined by Lapalme (2011), and the differences in design and alignment scope. The alignment approaches included in the main data source primarily gravitate towards the first school of thought (enterprise IT architecting) and the business-IT alignment scope. Due to its representation in terms of business-IT alignment scope, the contextualisation model is classified as a Business-IT Alignment Model (BIAM).

4.3 THE PROPOSED BUSINESS-IT ALIGNMENT MODEL (BIAM)

The purpose of this section is to relate the components of the BIAM to its theoretical foundations, followed by an in-depth discussion of every BIAM component. The section starts with a definition of the main BIAM components upfront to demonstrate the theoretical foundations of each component in section 4.3.1, followed by a detailed description of each component in section 4.3.2.

The results of the literature review indicated that business-IT alignment approaches provide answers to one or more of the following *three questions*:

- Question 1: '*Why* should the enterprise use the proposed approach to align?'
- Question 2: '*What* should the enterprise align?'
- Question 3: '*How* should the enterprise align?'

In answering the three questions through a conceptual mechanism, the BIAM subsequently consists of *four main components*:

- Component 1: An *alignment belief/paradigm of creating value* (Figure 45, foundation ellipse) (answering Question 1).
- Component 2: Three alignment *dimensions* (Figure 45, three panes of the block) to define the scope of alignment (answering Question 2).
- Component 3: Supporting *alignment mechanisms and practices* (Figure 45, bottom triangle) to ensure alignment across the alignment dimensions (partially answering Question 3).
- Component 4: *Alignment approach classifiers* that influences the selection of appropriate *alignment mechanisms and practices* (partially answering Question 3) (De Vries, 2010).

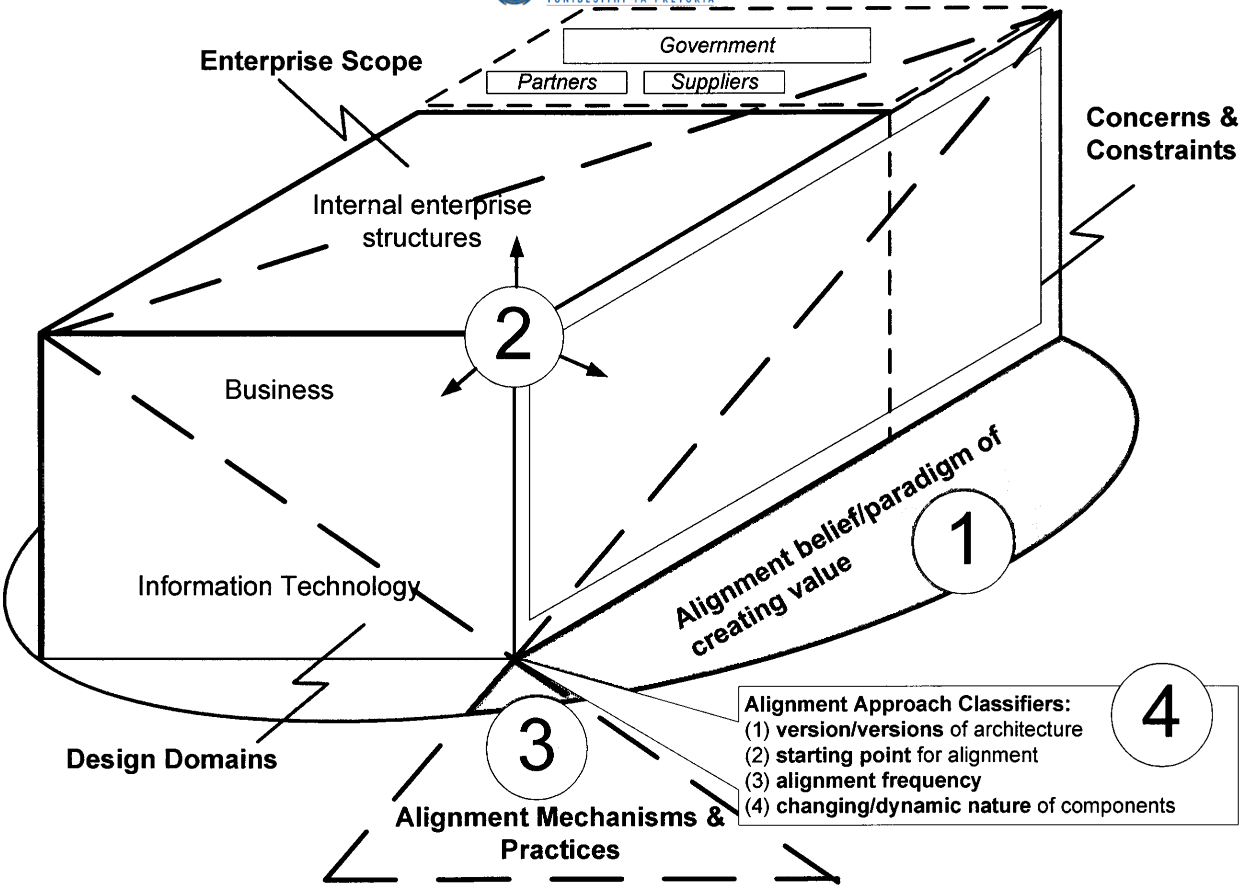


Figure 45: The BIAM (adapted from De Vries, 2010)

The core of the BIAM, is the *alignment mechanisms and practices* (Figure 45, Component 3 / bottom triangle), since they create the business-IT alignment *capability* that contributes to business-IT alignment. The alignment *mechanisms and practices* ensure alignment across Component 2, i.e. *design domains*, stakeholder *concerns*, and the *enterprise scope* (e.g. business units, programmes, and projects). Slicing across the three dimensions, the *alignment mechanisms and practices* thus form the core/heart of the BIAM, enacting alignment for the intended scope.

In support of the *alignment belief/value-creation paradigm* (Component 1) and three alignment *dimensions* (Component 2), the collective set of *mechanisms and practices* (Component 3) may be further characterised using *alignment approach classifiers* (Figure 45, Component 4 / callout). The classifiers relate to:

1. Version or versions of alignment (current state / future state)
2. Starting point for doing architecture work (top-down, bottom-up or middle in)
3. Alignment frequency (periodic vs continuous)
4. Different ways of addressing the changing/dynamic nature of the alignment components

The following sections relate the components of the BIAM to its theoretical foundations and delineate each component of the BIAM in terms of content and supportive literature sources.

4.3.1 Theoretical foundations supporting the BIAM

This section first provides an indication of how the current knowledge base was applied in constructing the BIAM (see Figure 46). A more detailed mapping is provided against the knowledge base during the discussion of every BIAM component in the subsequent sections.

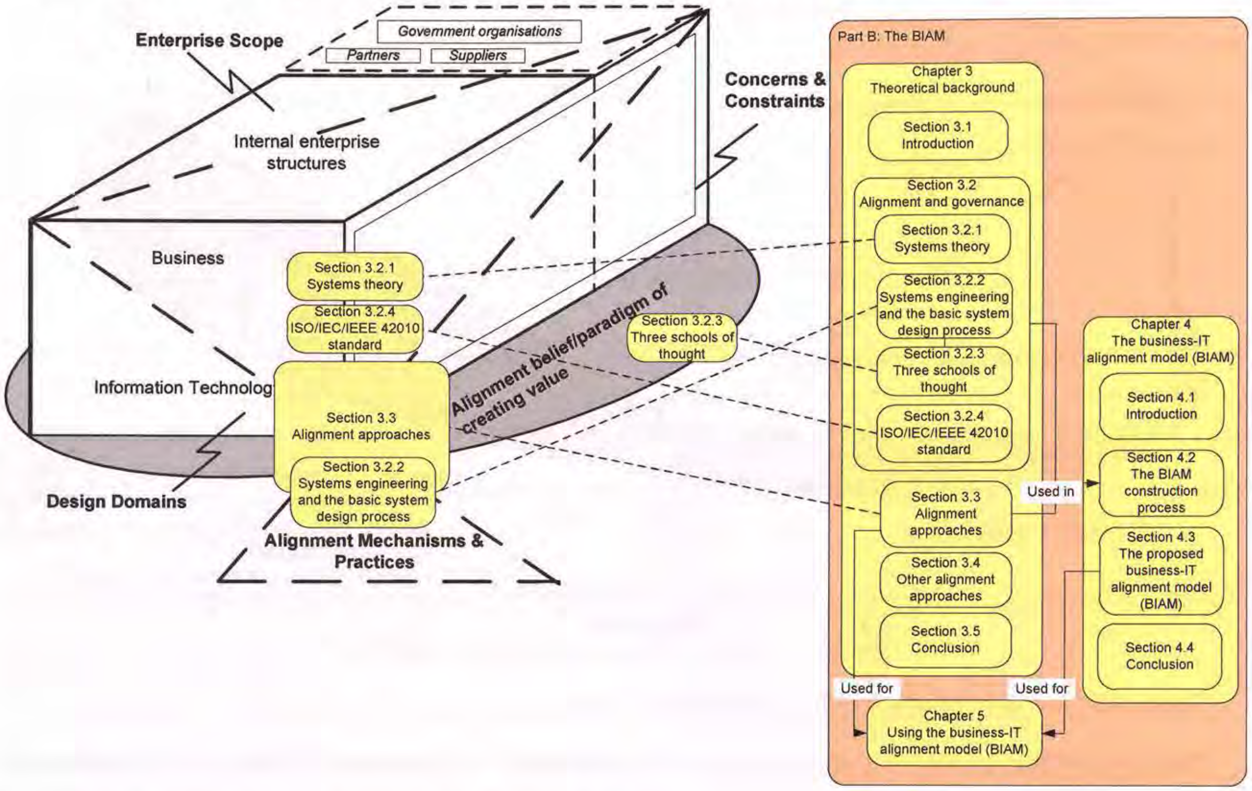


Figure 46: The theoretical foundations of the BIAM

Figure 46 illustrates that the foundation component of the BIAM (*Alignment belief/paradigm of creating value*) relates to the three different schools of thought and intends to accommodate different beliefs and paradigms, such as the three different schools of thought previously discussed in section 3.2.3.

Two of the three alignment *dimensions* (Figure 46, front and side panes, *Design Domains* and *Concerns & Constraints*) of the BIAM, represent the descriptive elements of the enterprise. Systems theory (covered in section 3.2.1) refers to different notions of a system and its representation using *white-box models* and *black-box models*, whereas the ISO/IEC/IEEE 42010 standard provides a standard for *architecture description* (covered in section 3.2.4).

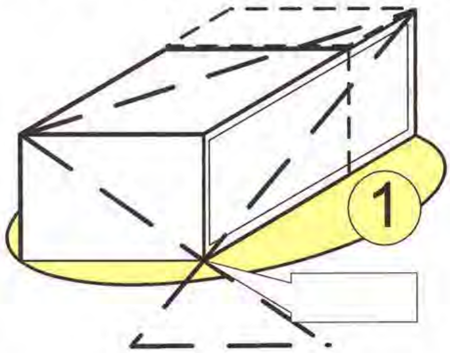
The *alignment mechanisms and practices* (Figure 46, *Alignment Mechanisms and Practices*) of the BIAM refer to the various means of alignment. Systems engineering and the basic system design process (covered in section 3.2.2) provides a systematic process to align different systems (e.g. the *organisation* of the enterprise system with ICT).

Finally, Figure 46 demonstrates that alignment approaches (covered in section 3.3) address one or more of the BIAM alignment components. As an example, the Open Group approach provides an *alignment belief/paradigm of creating value* (foundation component of the BIAM), delineates the scope of alignment in terms of three *alignment dimensions* (three panes of the BIAM block), and provides a rich set of *alignment mechanisms and practices* (bottom triangle of the BIAM).

4.3.2 The BIAM components

This section describes the various components of the BIAM according to Figure 45.

4.3.2.1 Component 1: Alignment belief/paradigm of creating value



The paradigm of value creation relates to the philosophical dimension of a paradigm, providing the *why* of the alignment approach and the grounds for the type of activities included in the alignment *mechanisms and practices*. Alignment approaches thus found their proposed approach on defendable value propositions/offerings. The value propositions are based on certain belief systems about value-creation in an enterprise and the capability of marketing the propositions to the owners/funding parties of the enterprise. Value is in the eye of the beholder (Hitchins, 2003), therefore alignment approaches differ in their value propositions. The value-proposition of an alignment approach is represented by the foundation component of the BIAM (Figure 45, Component 1).

Alignment approach author(s) provide a rationale for using a proposed alignment approach to address current miss-alignment problems in organisations. The authors, often influenced by their own worldview/epistemological beliefs, usually promise to deliver an alignment solution that will address the systemic miss-alignment causes in an enterprise. Similar to the different belief systems identified by Lapalme (2011) in terms of enterprise architecture, the three schools of thought could also be applicable to enterprise alignment:

1. Enterprise IT architecting
2. Enterprise integrating
3. Enterprise ecological adaption (see Table 8)

Although BIAM does not include or prescribe a taxonomy for classifying different schools of thought, BIAM acknowledges that a deeper paradigmatic analysis of alignment approaches would be useful as an extension of the BIAM, as discussed later in Chapter 12.

Enterprise Architecture (EA) is a discipline that could provide several means. The following themes emerge from various EA definitions in terms of its purpose/means:

- Some reason that EA needs to provide an *aggregate view* or a blueprint for directing the enterprise in terms of required high-level processes and IT capabilities (Boar, 1999; Ross et al., 2006; Winter & Fischer, 2007). Others (DeBoever, Paras, & Westbrock, 2010) also emphasise the intention of directing the enterprise on a strategic level; EA is described as a *strategic management discipline* that creates a *holistic view* of the business processes, systems, information, and technology. The *strategic management* focus will lead to more intelligent investment decisions, extending the life of assets and decrease the number of short term, high-cost implementations.

- According to ISO/IEC/IEEE 42010, architecture needs to create a *systems view*, i.e. the "fundamental organisation of a system, embodied in its components, their relationships to each other and the environment, and the principles governing its design and evolution" (ISO/IEC JTC 1/SC 7 committee, 2011). The components, their interaction and interrelationships, should be described in a consistent way to ensure holistic solutions in terms of the solution components (EA Research Forum, 2009; Handler, 2004; Lapkin, 2008; The Open Group, 2009; Theuerkorn, 2005; Winter & Fischer, 2007). A *systems view* should focus on reducing complexity of IT and business processes across the breadth of the enterprise, making a company more agile (DeBoever et al., 2010).

- Gartner (Willis, 2009, p. 7) reasons that EA is about the *continuous process of transformation* from a current architecture to a future architecture, i.e. "translating business vision and strategy into effective enterprise change" (Bernard, 2005; GAO, 2006; Lapkin, 2008; Schekkerman, 2004).

- Another prominent theme is *governance*, i.e. key principles that are required to govern the design and evolution of information systems, which impact various management areas such as maintenance, compliance, and risk management (Lapkin, 2008; The Open Group, 2009; Theuerkorn, 2005; Wagter et al., 2005; Willis, 2009; Winter & Fischer, 2007).

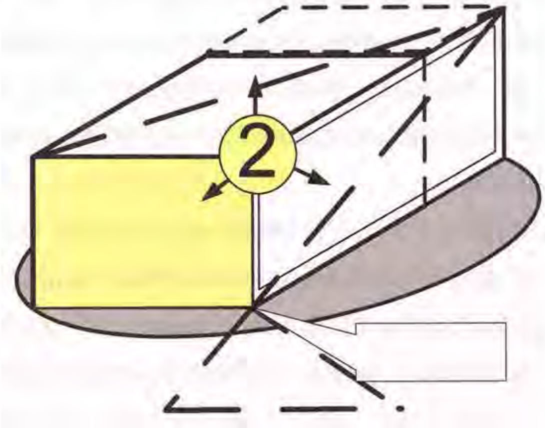
- A less prominent definition is that EA needs to provide an integrated and transparent *representation of all interests* and their current state of alignment. As interests of stakeholders constantly evolve, the representation of interests should also be constantly updated and reconciled. EA is thus an ongoing process (Sidorova & Kappelman, 2010).

Although the above-mentioned definitions reveal some of the value-creation means, practitioners still need to demonstrate value to the business in terms of bottom-line results. Alignment approaches thus need to demonstrate how the alignment approach will increase both efficiency and effectiveness (Buchanan & Soley, 2002; Rosser, 2004).

4.3.2.2 Component 2: Dimensions

BIAM depicts three *dimensions* (Figure 45, Component 2), depicted by the three panes of the block: design domains, concerns & constraints, and enterprise scope.

Design Domains (Figure 45, Component 2, front pane*)*



The first dimension provides the means for creating logical separation between different domains that require design.

Literature reveals many different conceptualisations for design domains. Hoogervorst (2009, p. 134) maintains that the demarcation/delineation of domains reveal "functional or constructional *system facets* for which *design activities* are required"; demarcation is *not simple* and requires specific system knowledge. Design domains may also be classified as *sub-systems* (for which *design activities* are required) if the sub-system parts interact with one another (Dietz, 2006). Defining the *boundary* of the sub-system is however contextual and depends on the intentions of the observer/analyst (Giachetti, 2010).

As an example of design domains, Winter and Fischer (2007) identified five domains: business, process, integration (e.g. enterprise services), software (e.g. software services and data structures), and technology/infrastructure. The Open Group (2009) defines slightly different design domains as part of TOGAF (The Open Group Architecture Framework): business, information system (which includes application and data), and technology. Hoogervorst (2009), focusing on enterprise alignment rather than business-IT alignment, defines four domains: business (the environmental system, customers requiring products/services), organisation (processes and employees), information and technology.

Taking the Zachman Framework as a second example, one may debate whether the Zachman Framework contains *design domains* or not. If one used the definition provided by Hoogervorst (2009, p. 134), one may reason that the six columns of the Zachman Framework (see Figure 24) are *system facets* for which *design activities* are required (i.e. inventory sets, process flows, distribution networks, responsibility assignment, timing cycles and motivation intentions).

Although different categorisation strategies exist for defining design domains, two broad categories of design domains emerge from our inductive research: *business* and *information technology*, which encapsulate more detailed design domains.

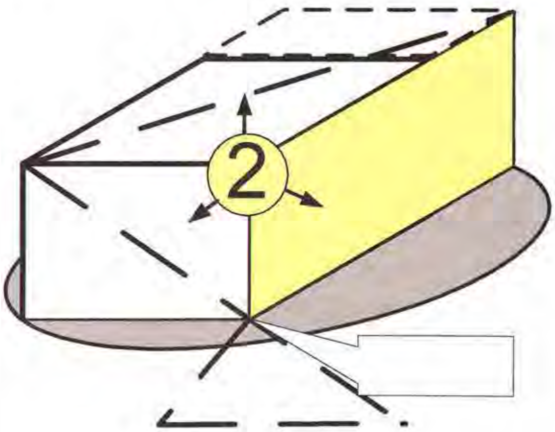
In understanding the *business* domain the following definitions on *business architecture* is used to describe the scope of the *business* domain:

- "Business architecture is a *general description of a system*. It identifies its purpose, vital functions, active elements, and critical processes and defines the nature of the interaction among them" (Gharajedaghi, 2006, p. 152).
- "It is a definition of what the *enterprise must produce* to satisfy its customers, compete in a market, deal with its suppliers, sustain operations, and care for its employees. It is composed of models of architectures, workflows, and events" (Whittle & Myrick, 2007, p. 31).
- "...business architecture is fitting the major elements of a business together"..."a set of *interrelated views of how a business works*" (McWhorter, 2008, p. 11). Supporting the latter, business architecture is "a *formal blueprint* of governance structures, business semantics, and value streams across the extended enterprise" (OMG's BAWG in Ulrich, 2008: 38).

In contrast, the *information technology* domain may consist of several layers that differ substantially amongst different frameworks. As an example, one could partition this domain into three sub-domains:

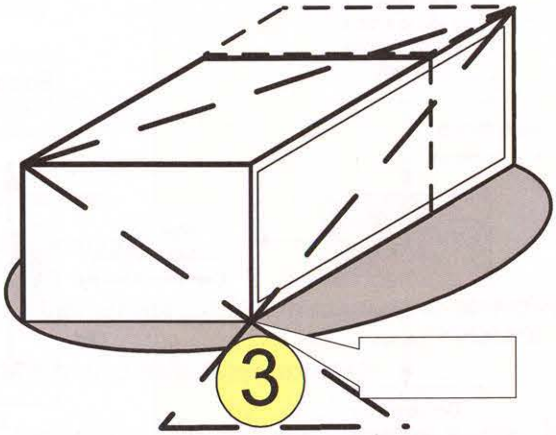
1. application (conveying the structure of specific applications, how they are designed, and how they interact with one another);
2. data (describing the logical and physical data stores in the enterprise); and
3. technical (describing the hardware and software infrastructure that supports applications and their interactions) (The Open Group, 2009).

Concerns and Constraints (Figure 45, Component 2, side pane*)*



The second dimension refers to *concerns and constraints* that should be addressed when the enterprise is designed. Different groups of stakeholders have a stake in enterprise performance, but are not necessarily in a position to influence performance (Gharajedaghi, 2006). The BIAM *concerns* (as depicted in Figure 45, *Component 2*, side pane) include those concerns that enterprise designers (e.g. enterprise managers, architects and engineers) would like to address during the design of the enterprise and its information systems. During the development of

4.3.2.3 Component 3: The alignment mechanisms and practices



The set of applicable alignment *mechanisms and practices* (Figure 45, Component 3) that supports a specific alignment approach depends on the *alignment belief/paradigm of creating value* (Figure 45, Component 1) and the alignment strategy that enables alignment across the relevant alignment *dimensions* (Figure 45, Component 2).

In practice, *alignment mechanisms and practices* are usually organised as an integrated *set of alignment mechanisms and practices* as part of a methodology. TOGAF ADM (architecture development methodology) is an example of a *methodology*, which includes nine sequential and/or iterative phases and numerous mechanism and practices. Hoogervorst (2009, pp. 221, 316) also suggests an alignment *process* to enact alignment on different levels of scope.

The set of *alignment mechanisms and practices* focuses on different levels of alignment scope, depending on the object system that needs to be constructed, i.e. either the ICT system or the enterprise system. Figure 47 illustrates the different levels of alignment scope addressed by a set of *alignment mechanisms and practices*. The *enterprise system design process* starts with knowledge about the construction of the using system, i.e. the environmental system (government, regulations, industry, markets, competitors etc.), which is necessary to determine the functional requirements for the object system, i.e. the enterprise system (see Figure 47). The functional requirements specify the products/services that need to be delivered, and the customers/markets that will be served. Although functional requirements determine largely the construction of the enterprise (i.e. integrated processes, skills and technology competencies), non-functional requirements (e.g. flexibility, cost, security, cultural-impact etc.) also determines/constrains the construction of the enterprise.

The basic system design process (see Figure 47) also provides a reference to relate *strategic choices* (as defined by Hoogervorst, 2009) to functional changes and constructional changes in the enterprise.

The colours used in Figure 47 are meaningful. The light shade of yellow demonstrates alignment when designing an ICT system, which applies to BIAM during the contextualisation of current alignment approaches in Chapter 5. The bright yellow demonstrates alignment when designing the enterprise system as the object system.

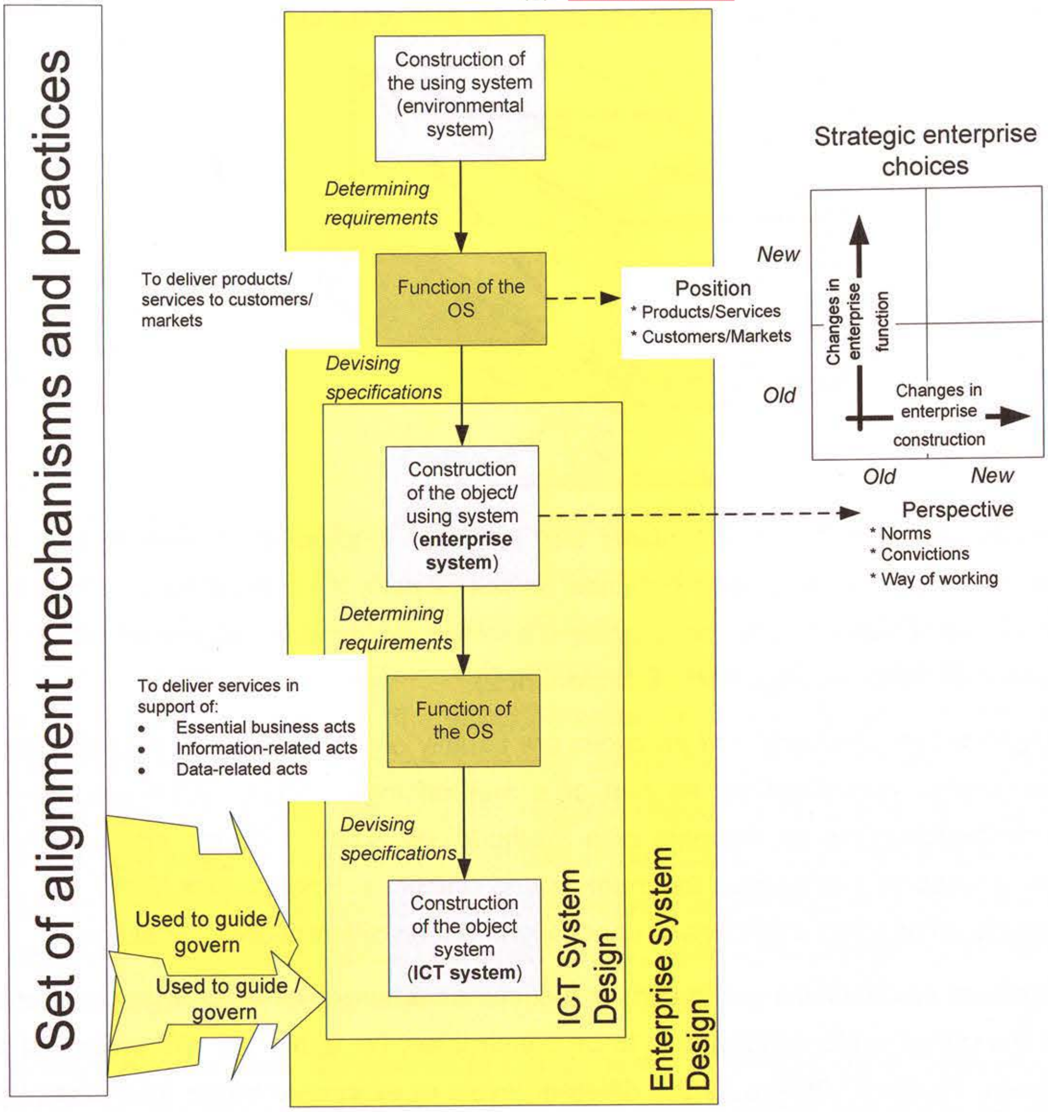


Figure 47: Relationships between a set of alignment mechanisms and practices, the system design process, and enterprise strategic choices

By applying appropriate *alignment mechanisms and practices* in an enterprise, the enterprise has the potential for creating an *enterprise alignment/governance competence,* i.e. "the organisational competence for continuously exercising guiding authority over strategy and architecture development, and the subsequent design, implementation and operation of the enterprise" (Hoogervorst, 2009, p. 265).

The following list of mechanisms and practices is neither integrated nor exhaustive, but rather an example of *alignment mechanisms and practices* found in literature. The list of mechanisms and practices all relate to the BIAM mechanisms and practices (Figure 45, Component 3) and include:

1. Architecture description and reference models
2. Alignment/design methodologies
3. Architecture principles and standards

4. Additional management mechanisms and practices
5. Governance frameworks
6. Transformation roadmaps
7. Analyses (e.g. gaps/impact)
8. Maturity models
9. Skills/learning requirements
10. Software tools and/or guidance

The remainder of this section delineates the ten *mechanisms and practices* categories.

1. Architecture description and reference models

A consistent architecture description contributes towards unity, integration and alignment. According to the ISO/IEC/IEEE 42010 standard (ISO/IEC JTC 1/SC 7 committee, 2011), an *architecture description* is a "work product used to express an architecture". An example of a work product that expresses architecture of an enterprise, is the content metamodel of TOGAF (The Open Group, 2009).

Although BIAM is not normative in terms of the elements of an *architecture description*, the terminology aligns with the descriptions provided by ISO/IEC/IEEE 42010. Table 11 relates the components of BIAM to the elements of *architecture description* provided by the ISO/IEC/IEEE 42010.

Table 11: BIAM components related to ISO/IEC/IEEE 42010 architecture description components

BIAM components / sub-components	**ISO/IEC/IEEE 42010 architecture description components**
(1) Alignment belief/paradigm of value creation.	No direct mapping.
(2) Dimensions	
Dimension 1: Design domains.	May be similar to *viewpoints* if the *viewpoints* are *facets* that require design.
Dimension 2: Concerns.	*Concerns*.
Dimension 3: Enterprise scope.	No direct mapping.
(3) Alignment mechanisms and practices	
Architecture description	*Architecture description.*
…	
Other mechanisms and practices.	No direct mapping.
(4) Alignment approach classifiers.	No direct mapping.

Architecture frameworks and *architecture description languages* use elements of the complete *architecture description* (see section 3.2.4).

Numerous EA *frameworks* exist, for example the Zachman Framework, TOGAF (the Open Group Architecture Framework), IAF (Integrated Architecture Framework), E2AF (Extended Enterprise Architecture Framework), PERA (Purdue Enterprise Reference Architecture), CIMOSA (Computer Integrated Manufacturing Open System Architecture), FEAF (Federal Enterprise Architecture Framework), JTA (Joint Technical Architecture), and DODAF (Department of Defence Architecture Framework) (Schekkerman, 2004). However, not all of these frameworks conform to the ISO/IEC/IEEE 42010 standards on defining *architecture frameworks*.

Frameworks may be associated with *languages*. Examples include BPMN (Business Process Modelling Notation), IDEF (Integrated Definition Language), UML (Unified Modelling Language), and ARIS (Architecture of Integrated Information Systems). Not all of these *languages* however conform to the ISO/IEC/IEEE 42010 standards for defining *architecture description languages*.

Generic *reference models* may be used to quick-start architecture efforts, re-use previous architectures, optimise according to best-practice reference models, and/or ensure integration across *design domains*. TOGAF (The Open Group, 2009) provides *reference models* across an enterprise continuum that ranges from a set of generic foundation architectures to enterprise-specific architectures. Various classifications can be used to partition and organise the enterprise continuum, e.g. subject matter (products, services) and viewpoint (functional breakdown or *design domain* breakdown). Other examples or *reference models* include GERA (Generalised Enterprise Reference Architecture), SCOR (Supply Chain Operations Reference model), VCOR (Value Chain Operations Reference), and e-TOM (Enhanced Telecom Operations Map) for business processes in the telecommunications industry. More examples include TRM (Technical Reference Model) and III-RM (reference model for integrated information infrastructure) developed by The Open Group. The OMB (2007a) provides reference models for every *design domain*, i.e. performance reference model, business reference model, service component reference model, technical reference model and data reference model.

2. Alignment/design methodologies

A *methodology* is a phased problem-solving approach, usually following a general problem-solving methodology:

1. scoping the problem,
2. designing the solution,
3. evaluating the solution, and
4. re-visiting the problem if the solution is unsatisfactory (Giachetti, 2010).

Alignment/design methodologies are often used to encapsulate other *alignment mechanisms and practices*. An example of a methodology is the TOGAF ADM (architecture development methodology). Hoogervorst (2009, p. 221) does not explicate a methodology to enact alignment, but also implies a *process* to create enterprise alignment.

Depending on the level of alignment (ICT developments or the development of the entire enterprise), alignment/design methodologies guides the *design process* of either the ICT system or the enterprise system (see Figure 47). Hoogervorst (2009, p. 262) emphasises that the design of the enterprise and its ICT system often occurs concurrently.

A number of *alignment/design methodologies* exist for designing the ICT system, e.g. Rapid Application Development (RAD), Architected Rapid Application Development (Architected RAD), Dynamic Systems Development Methodology (DSDM), Joint Application Development (JAD), Information Engineering (IE), Rational Unified Process (RUP) and Structured Analysis and Design (SAD) (Whitten & Bentley, 2007). Although a number of publications address the *importance of design* in enterprises (Giachetti, 2010; Hammer & Champy, 1993; Johansson, McHugh, Pendelbury, & Wheeler, 1993; Martin, 1995; D. A. Nadler & Tushman, 1997), few or over-simplified enterprise design/engineering *methodologies* exist, possibly due to the complexity of the enterprise and the multiple stakeholders involved. Also, the emphasis in literature is on *enterprise management* (the functional perspective on the enterprise), rather than on *enterprise design* (the constructional perspective of the enterprise) (Hoogervorst, 2009).

3. Architecture principles and standards

Architecture principles are general rules and guidelines that supports the way in which an enterprise intends to fulfil its mission (The Open Group, 2009, p. 265). Hoogervorst (2009, p. 127) argues that *principles and standards* ensure a unified and integrated design, addressing multiple concerns. Although TOGAF (The Open Group, 2009) provides examples of principles for every *design domain*, Hoogervorst (2009) states that some architecture principles or standards may apply to more than one *design domain* and address more than one *concern*. The practical distinction between *principles* and *concerns* is sometimes blurred, as some functional concerns may be generic to a class of systems and thus adoptable as principles, rather than concerns (Hoogervorst, 2009).

An example of a set of *standards* is the SIB (Standards Information Base) of TOGAF, which is a catalogue of technology standards and specifications that are useful in implementing the services identified in the TRM (Technical Reference Model).

4. Additional management mechanisms and practices

Several mechanisms and practices are included for management areas (e.g. architecture management, strategy management, risk management, change management, project management, and program management; on both an enterprise management level and IT management level) to ensure coherency and consistency (Hoogervorst, 2009; The Open Group, 2009). Examples of architecture management mechanisms include architecture boards/committees, architecture compliance reviews at pre-defined project milestones/checkpoints, architecture compliance review checklists and guidelines (Ross et al., 2006; Schekkerman, 2006; The Open Group, 2009; Weill & Ross, 2004).

5. Governance frameworks

Governance frameworks provide a collection of required areas to yield effective governance (Hoogervorst, 2009). Frameworks that are often mentioned include CobiT (Control Objectives for Information and related Technology), ITIL (IT Infrastructure Library) and ISO 17799 (Symons, 2005). According to the Open Group (2009) CobiT is a good source of information on IT governance. Hoogervorst (2009) however reasons that neither COBit, nor ITIL, nor ISO can be classified as governance frameworks. He argues that CobiT is a framework for IT *management* (containing a large number of IT management tasks, rather than governance practices that guide design), whereas ITIL is a set of best practices for *IT service management*, and ISO only directs security issues.

6. Transformation roadmaps

DeBoever et al. (2010) maintain that roadmaps are the primary output of enterprise architecture. The roadmaps list individual increments of change according to a timeline to show progression from the *current state* to *future state* business processes, systems, information and technology. Transformation roadmaps and practices are common to frameworks such as IAF, GERAM and TOGAF.

7. Analyses (e.g. gaps/impact)

The purpose of analysing architecture components and their relationships is to identify performance *gaps* or *gaps* between the current-state architecture and future state architecture. The analyses of proposed future-state architecture could also highlight the *impacts* of the future-state architecture on existing architecture components. The analyses are often used as change drivers, guiding decision-making related to the evolution of architectures (Dunshire, O'Neill, Denford, & Leaney, 2005; The Open Group, 2009).

8. Maturity models

Maturity models measure alignment/governance capabilities at an enterprise. Examples include the ACMM (Architecture Capability Maturity Model) developed by the US Department of Commerce (The Open Group, 2009), the Federal Enterprise Architecture Program EA Assessment Framework 2.0 (OMB, 2005), the SAM (Strategic Alignment Maturity) model of Luftman & Kempaia (2007), used to indicate IT-business alignment maturity, and the eight dimensions of EA maturity advanced by the Gartner Group (James & Burke, 2005). Distinguishing between two levels of alignment, Hoogervorst (2009) provides two maturity models, an IT governance maturity model and an enterprise governance maturity model.

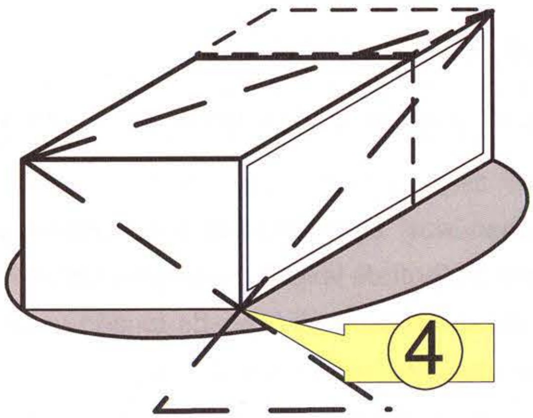
9. Skills/learning requirements

An alignment approach requires employees and *personal competencies* to apply suitable *alignment mechanisms and practices*. According to Hoogervorst (2009) the enterprise architect needs to master several topics within six areas (systems thinking, business and organisation, information, IT, enterprise development and change, and general topics). Ross et al. (2006) define different skill sets for CIO's (chief information officers) based on the maturity level of the enterprise. The Open Group (2009) provides an EA skills framework to define sets of generic skills, business skills and methods, enterprise architecture skills, program and project management skills, IT general knowledge skills, technical IT skills, and legal environment skills. Different skill levels (level 1 to 4) per skill, apply for different architecture roles (e.g. architecture board member, architecture sponsor, EA manager etc.).

10. Software tools and/or guidance

This mechanism includes the wide variety of tools and tool sets that are available for designing various architecture artefacts. Examples include the Systems Architect Family, ARIS Process Platform, the Metis Product Family, and ABACUS. Schekkerman (2011) provides a comparisons of enterprise architecture tools, whereas TOGAF (The Open Group, 2009) provides evaluation criteria and guidelines choosing automated tools.

4.3.2.4 Component 4: Alignment approach classifiers



BIAM provides four classifiers to differentiate between alignment approaches in 'how' they ensure alignment (Figure 45, Component 4). The BIAM foundation (*alignment belief/paradigm of creating value*) directly influences the alignment approach, which in turn influences the set of *alignment mechanisms and practices* that are required in combination with the alignment approach. The four alignment approach classifiers are:

1. Version/versions of architecture
2. Starting point for alignment
3. Addressing the dynamic nature of architecture components
4. Periodic vs. continuous alignment

The remainder of this section delineates the four *alignment approach classifiers*.

1. Version/versions of architecture

The version of alignment refers to the version of the architecture blueprints with reference to the *design domains* and *concerns*. Alignment approaches differ in their focus on creating current and/or future versions of architecture.

Some alignment approaches focus on building a complete blueprint of the current (*as-is*) architecture. These theoretical models analyse the current architectures before starting the future architectures. The Open Group (2009) in its ADM (Architecture Development Method) follows a systematic process in analysing current architectures in defining gaps (gap analyses). The rationale is that a current architecture would highlight inefficiencies, reveal opportunities for centralisation, and lead to cost-cutting efforts.

Other alignment approaches focus on the future (*to-be*) architectures, while following a pragmatic approach in building a sub-set of *as-is* architectures, depending on the purpose of the architecture exercise, e.g. providing a baseline for developing a transition strategy. Detailed modelling is only conducted in a selected and highly pragmatic way (Buchanan & Soley, 2002; DeBoever et al., 2010; Lapkin, 2008), based on the principle of *just enough architecture, just in time*.

2. Starting point for alignment

Alignment approaches either propose a top-down or bottom-up approach in developing *design domains*.

Some alignment approaches start at strategy and the business domain (top level), working towards the technical domains (bottom levels). Examples include TOGAF ADM and the Gartner EA Process model. The rationale is that EA needs to add value in terms of the strategy and business-operation of the enterprise.

As an alternative, design could also start at the technology domains (bottom levels). The rationale for starting at the bottom is that a flexible IT infrastructure would easily accommodate changes in the business domains. SOA (Service Oriented Architecture) projects are based on this paradigm (Robertson, 2005). According to The Open Group SOA Working Group (2007, p. 9), "a major benefit of SOA is that it delivers enterprise agility, by enabling rapid development and modification of the software that supports the business processes – and hence makes it easier to change the business processes themselves". Hoogervorst (2009, p. 105) uses the word *enablement* to describe the bottom-up approach. He maintains that enterprises should not only create IT-arrangements, but rather enterprise arrangements that would enable new emerging enterprise strategies. The rationale is that strategy development often does not follow a linear, analytical top-down pattern, but follows an incremental, evolutionary development process (Ciborra, 2002), derived from the complex set of business, competitive, organisational and environmental circumstances (Weill & Broadbent, 1998).

Locke (2009a, p. 79) also reports on another approach, called the middle-in approach. The middle-in approach refers to distinct *concerns* (Figure 24, six rows of the Zachman Framework) associated with the enterprise design process, e.g. scope contexts (executive perspective), business concepts (business management perspective), *system logic* (architect perspective), technology physics (engineer perspective), tool components (technician perspective), and operations instances (enterprise perspective). The rationale is that implementation of an ERP (enterprise resource planning) system, requires a middle-in approach, starting at the *system logic* level, working both 'up' and 'down' the design process to implement the system.

3. Addressing the dynamic nature of architecture components

Zachman (1996) considered the usefulness of EA when observing the architecting effort required for a Boeing 747 aircraft (Zachman, 2009b). However, the inherent design of an aircraft changes relatively slowly over time. One of the *typical system propertie*s of an enterprise is its *dynamic* nature (see 3.2.1) Enterprise design does not occur at a single point in time, as enterprises evolve over time and are constantly changing (Giachetti, 2010). *Dynamics* are at the heart of regulation in organismic systems, rather than control and feedback (Hitchins, 2003). Alignment approaches propose different means for addressing the dynamic nature of architecture components.

The Open Group (2009) maintains that the practice of open standards and boundaryless integration across departmental/divisional/enterprise boundaries address the challenges associated with dynamic changes. The rationale is that maximum flexibility through design creates the ability to change swiftly. However, alignment across the supply chain, integrating diverse databases and applications written in different languages remains a challenge. Different integration languages partially address the language challenge, e.g. DCOM (Distributed Component Object Model), CORBA (Common Object Request Broker Architecture), Enterprise JavaBeans, and XML (Extensible Markup Language). Object-orientated and service-orientated design approaches also attempt to ensure flexibility via loosely-coupled components that could easily be re-used or assembled in a make-to-requirement fashion.

Some alignment approaches acknowledge that technical architecture design practices could create flexibility, but emphasise governance practices that are required to enact change (Bittler & Kreizman, 2005; Wagter et al., 2005).

4. Periodic vs continuous alignment

Alignment approaches often reveal different paradigms regarding alignment frequency. Some models promote once-off alignment endeavours. The models are supported by the analysis of current and future architectures to identify gaps, which may lead to rip-and-replace efforts, e.g. BPR (Business Process Re-engineering) (Whitten & Bentley, 2007).

Other models address systematic alignment that is part of an ongoing, incremental enterprise design activity (Giachetti, 2010). BPM (business process management) is an example of an ongoing process of aligning business requirements with information system functionality and its supporting infrastructure (Whitten & Bentley, 2007). The rationale is that an incremental approach, i.e. creating alignment one project at a time, produce quick wins to create credibility (DeBoever et al., 2010; Ross et al., 2006).

The alignment approach classifiers of the BIAM (Figure 45, component 4) thus provide four classifiers to differentiate between alignment approaches in 'how' they ensure alignment, i.e. focusing on different versions of architecture, different starting points for alignment, addressing the dynamic nature of architecture components, and using different frequencies of alignment.

4.4 CONCLUSION

Chapter 4 recognized the knowledge embedded in current alignment approaches and used exploratory design and a literature review to inductively create a Business-IT Alignment Model (BIAM). The inductive process highlighted prominent themes/patterns evident in current alignment approaches.

The chapter delineated how BIAM answers *three* questions using *four* BIAM components. The *three* questions are:

- Question 1: '*Why* should the enterprise use the proposed approach to align?'
- Question 2: '*What* should the enterprise align?'
- Question 3: '*How* should the enterprise align?'

The *four* BIAM components are:

- Component 1: An *alignment belief/paradigm of creating value.*
- Component 2: Three alignment *dimensions* to define the scope of alignment.
- Component 3: *Alignment mechanisms and practices* to ensure alignment across the alignment dimensions.
- Component 4: *Alignment approach classifiers* that influences the selection of appropriate alignment mechanisms and practices.

The next chapter (Chapter 5) uses the BIAM to compare and contextualise two prominent alignment approaches (the Zachman approach and the Open Group approach). Later, Chapters 7 and 8 also use the BIAM to compare and contextualise two less prominent alignment approaches (the *foundation for execution* approach and the *essence of operation* approach).