Executive summary

With the appearance of Version 2.0, the ArchiMate standard for enterprise architecture modelling now offers full modelling support throughout the architecture development cycle. It allows for the description and visualization of different architecture domains and their relationships, as well as the motivation behind the architecture, and the implementation projects and migration planning. ArchiMate complements TOGAF, the standard method of The Open Group for developing enterprise architectures. In this paper, we highlight what is new in the new versions of these two standards, and we illustrate how they can be used together to provide an integral approach to enterprise architecture.
## Contents

**Introduction** ................................................................................................................................. 3  
**An Integrated Approach to Enterprise Architecture** ........................................................................ 3  
**Way of Working: TOGAF** .................................................................................................................. 4  
  - The components of TOGAF .................................................................................................................. 4  
  - What's new in TOGAF 9.1? ................................................................................................................... 4  
  - The Architecture Development Method (ADM) .................................................................................... 5  
**Way of Modelling: ArchiMate** .......................................................................................................... 6  
  - What's new in ArchiMate 2.0? .............................................................................................................. 6  
  - Summary of the ArchiMate core .......................................................................................................... 7  
  - Summary of the ArchiMate extensions .................................................................................................. 7  
  - ArchiMate and TOGAF ....................................................................................................................... 9  
**Example case: ArchiSurance** ............................................................................................................. 9  
  - Motivation ............................................................................................................................................ 10  
  - Business Architecture .......................................................................................................................... 10  
  - Application Architecture ..................................................................................................................... 11  
  - Data Architecture ............................................................................................................................... 12  
  - Technology Architecture .................................................................................................................... 13  
  - Gap Analysis ....................................................................................................................................... 13  
  - Implementation and migration planning .............................................................................................. 14  
**Summary and Conclusions** ............................................................................................................... 15  
**References** ......................................................................................................................................... 16
Introduction

Early in 2012, Version 2.0 of ArchiMate, the enterprise architecture modelling standard of The Open Group, was published. A few months before that, the 9.1 update of TOGAF came out, the standard framework and method for enterprise architecture development of The Open Group.

ArchiMate was developed with the aim to provide a uniform representation for enterprise architecture descriptions [4][5][6]. It offers an integrated architectural approach by which organizations can describe and visualize different architecture domains, as well as their underlying relationships and dependencies. In Version 2.0 of the language, additional concepts have been added to model the motivation behind the architecture, and to support implementation and migration planning. With these extensions, ArchiMate provides full modelling support throughout the TOGAF architecture development cycle.

In this paper, we outline an integrated approach to enterprise architecture based on these two standards of The Open Group. We briefly introduce TOGAF and ArchiMate, and highlight what is new in the new versions of these standards. We illustrate our approach with an example.

An Integrated Approach to Enterprise Architecture

Frameworks for enterprise architecture vary in the types of support that they offer. They may have, among others, any combination of the following ingredients (see Figure 1):

- A process ("way of working") for creating architectures; this may be accompanied by guidelines, techniques and best practices.
- A set or classification of viewpoints.
- A language ("way of modelling") for describing architectures (defining concepts and relationships, but also a notation).
- The concept of a (virtual) architecture repository, possibly containing predefined architectural artefacts and (reference) models.

![Figure 1. Ingredients of an Integrated Enterprise Architecture Approach](image)

TOGAF and ArchiMate have a firm common foundation in their shared notion of enterprise architecture. Both adopt the central concept of viewpoints on a single underlying model repository, aimed at a specific set of stakeholders and concerns. On the other hand, the standards complement each other: TOGAF provides an elaborate method, including a process, guidelines and techniques, for enterprise architecture development, while ArchiMate provides a well-defined language, including a graphical notation, for enterprise architecture modelling. Together, these two standards make up a complete and integrated approach to enterprise architecture.
Way of Working: TOGAF

The components of TOGAF

The main structure of TOGAF 9.1 consists of six components:

1. The core of TOGAF is formed by the Architecture Development Method (ADM), a step-wise iterative approach for the development and implementation of an Enterprise Architecture. The phases of the ADM will be explained in some more detail below.

2. A collection of Guidelines and Techniques to support the application of the ADM. The guidelines address adaptation of the ADM in order to deal with a number of usage scenarios, including different process styles (for example, the use of iteration) and specific specialist architectures (for example, the security architecture). The techniques support specific tasks within the ADM, such as, defining principles, business scenarios, gap analysis, migration planning and risk management.

3. The Architecture Content Framework (ACF) provides a detailed model of architectural work products, including deliverables, artefacts within deliverables and the architecture building blocks represented by deliverables. While the ACF defines a content metamodel that specifies the relevant types of building blocks, it does not provide a complete modelling language; this is where ArchiMate complements TOGAF.

4. The Enterprise Continuum describes a view of the architecture repository (a concept which is also worked out in TOGAF), and provides methods for classifying architecture and solution artefacts, showing how the different types of artefacts evolve, and how they can be leveraged and reused. This is based on architectures (in the Architecture Continuum) and solutions (in the Solutions Continuum) existing both in the enterprise and in the industry at large, which the enterprise has collected for use in the development of architectures.


6. The Architecture Capability Framework is a set of resources, guidelines, templates and background information provided to help the architect to establish an architecture practice within the organization.

What's new in TOGAF 9.1?

TOGAF 9.1 is a maintenance update to TOGAF 9, and does not contain many substantial changes compared to its predecessor (unlike the major upgrade from Version 8 to Version 9 in 2009). There are several editorial improvements and corrections throughout the document. The objectives of all the ADM phases have been reformulated, and the descriptions of the implementation- and migration-related phases have been extended. Also, some pieces have been removed from the standard (e.g., the document classification model and the building block example).
The Architecture Development Method (ADM)

The core of TOGAF is formed by the Architecture Development Method (ADM), a step-wise, iterative process for the development and implementation of an enterprise architecture (see Figure 2).

The ten phases of the ADM can be grouped into four main parts, also shown in Figure 2:

7. “Getting the organization committed and involved”: the Preliminary Phase prepares the organization as a whole for “working under architecture”, and involves activities such as establishing an architecture capability, tailoring the architecture methods and techniques for the specific characteristics of the organization, and defining an initial set of architecture principles; Phase A, Architecture Vision, prepares for a single architecture development cycle, and includes the formulation of an architecture vision with a high-level overview of the change that is envisaged.

8. “Getting the architecture right” concerns the description of the actual baseline and target architectures, and an analysis of the gaps between baseline and target. The three phases in this group are concerned with three main types of architectures: business architecture (Phase B), information systems architectures (Phase C), and technology architecture (Phase D).

9. “Making the architecture work” is concerned with the implementation of the developed architecture, and planning the migration to the new situation. It includes Phase E, Opportunities and Solutions, in which the gap analysis results are consolidated and potential implementation work packages are identified; Phase F, Migration Planning, in which work packages are prioritized and a migration plan is established; and Phase G, Implementation Governance, safeguarding the compliance of implementation projects with the architecture.

10. “Keep the process running” is concerned with the management, prioritization and version control of requirements on the architecture. The central Requirements Management process manages the requirements during an architecture development cycle. In Phase H, Change Management, new requirements are identified that may lead to the start of the new architecture development cycle.

TOGAF also includes the identification of viewpoints, techniques and reference models. However, it does not define an actual modelling language. The TOGAF Architecture Content Framework does indeed identify relevant architecture building blocks, but it does not constitute a precisely defined language, nor does it provide a notation for these building blocks. ArchiMate complements this by defining a fully worked out (graphical) modelling language, including the definition of relevant viewpoints. This language also provides a concrete visualization of the views identified in TOGAF.
Way of Modelling: ArchiMate

Within larger organizations, one can typically find various architecture domains, such as organizational structures, products, business processes, information systems, applications and technical infrastructure. Traditionally, each architecture domain employs specific models and visualizations, which simplifies communication, discussion and analysis within the domain. However, the relations between these different domains are in many cases unclear. Moreover, these domains tend to (at least partially) overlap. Therefore, ArchiMate provides a unified way to model enterprise architectures, while integrating the various domains and describing them in an easily readable way, as illustrated in 0.

ArchiMate is positioned at the level of enterprise architecture, which implies that the ArchiMate language does not provide the level of detail one would typically find in languages used at the “design level” [10], such as BPMN [9] for business process design and UML [8] for application and technical infrastructure design.

ArchiMate has been designed in a structured way, by defining a generic structure that is made specific for the different architectural layers. It also defines a limited set of relation types that are used throughout the language. Finally, ArchiMate provides a standard graphical notation for the modelling concepts and relations.

In this section, we briefly discuss the structure and main elements of the ArchiMate language. In [10] a more detailed account is provided of the requirements on the language, and the design decisions that underpin its design.

What’s new in ArchiMate 2.0?

In the ArchiMate core, a large number of minor improvements have been made compared to ArchiMate 1.0: inconsistencies have been removed, examples have been improved, and additional text has been inserted to clarify certain aspects. Two new concepts have been added to the core, based on needs experienced by practitioners:

- **Location**, to model a conceptual point or extent in space that can be assigned to structural elements and, indirectly, of behaviour elements.
- **Infrastructure function**, to model the internal behaviour of a node in the technology layer. This makes the technology layer more consistent with the other two layers.

Furthermore, ArchiMate 2 adds two extensions to the language. With these extensions, ArchiMate provides modelling support throughout the TOGAF ADM:
• The **Motivation extension** defines concepts to model the motivation for the choices made in the design of the architecture. This includes concepts such as stakeholder, driver, goal, requirement and principle. For motivation elements, a limited set of relationships has been defined, partly reused from the ArchiMate core.

• The **Implementation & Migration extension** defines concepts to support the identification of implementation projects and migration planning. This includes concepts such as work package, deliverable, plateau and gap.

Simultaneously with the publication of ArchiMate 2.0, The Open Group also launched a certification programme for ArchiMate practitioners, tools and training courses.

**Summary of the ArchiMate core**

The ArchiMate core defines the concepts to model actual architectures, as described in Phases B, C and D of the TOGAF ADM. This is essentially the part of ArchiMate as described in Version 1.0 of the language, with the exception of the changes mentioned before. Figure 4 summarizes the most frequently used concepts of the ArchiMate core, positioned within the ArchiMate framework. The framework identifies three architectural layers, which largely correspond to the three phases for architecture development in the TOGAF ADM: the *business* layer, the *application* layer, and the *infrastructure* layer. Within each layer, three aspects are described: the *active structure*, i.e., the entities that perform behaviour (e.g., business actors, application components, infrastructure nodes), the *behaviour* (e.g., processes, functions, and services), and the *passive structure*, i.e., the (information) objects that are processed as part of the behaviour.

**Summary of the ArchiMate extensions**

As described in the previous sections, the ArchiMate core (and therefore, ArchiMate 1.0) chiefly supports modelling of the architectures in Phases B, C and D in the TOGAF ADM (“Getting the architecture right” in Figure 2), as is illustrated in Figure 13. The resulting models are used as input for the subsequent ADM phases; modelling concepts specifically aimed at the other phases – e.g., concepts for modelling principles, goals and requirements, or concepts to support migration planning – were still missing in Version 1.0 of the language. However, as stated before, ArchiMate 2.0 provides two extensions for this: one for describing motivation (e.g., stakeholders, goals and requirements), supporting the phases for “Getting the organization committed and involved” and “Keep the process running” from Figure 2, and one for implementation and migration planning, supporting the phases for “Making the architecture work” from Figure 2. The next subsections outline these two extensions.
Figure 5 adds the two extensions to the ArchiMate framework, and shows how the main concepts from the extensions are related to the core.

The motivation extension of ArchiMate [1] adds concepts related to goal modelling and business requirements management. They can be used for the identification, description, analysis and validation of goals and requirements at business level and their realization in enterprise architecture models as described with the ArchiMate core concepts. The motivational concepts, based on sources such as OMG’s Business Motivation Model [11], architecture principles [12][13] and goal-driven requirements engineering [14][15][16] are used to model the motivations, or intentions, that underlie the design of an enterprise architecture. These intentions influence, guide and constrain the design. Intentions are pursued by stakeholders, which can be individuals or groups such as a project team, enterprise or society. In addition, intentions may be organized into certain areas of interest, called drivers, such as customer satisfaction, compliance to legislation or profitability.

The actual intentions are represented by goals, principles and requirements. Goals represent some desired result – or end – that a stakeholder wants to achieve; e.g., increasing customer satisfaction with 10 percent. Principles and requirements represent desired properties of solutions – or means – to realize the goals. Principles represent desired properties that are required from all possible solutions in a given context; requirements represent desired properties of specific, individual solutions. For example, a requirement “Use a single CRM system” is a specialization of a principle “Data should be stored only once” by applying it to the current organization’s architecture in the context of the management of customer data. A constraint is a specific type of requirement that restricts the way in which a system is realized. Requirements are realized by core elements.

The Implementation and Migration Extension defines a number of additional concepts that enable the modelling of the architecture change process and increase the insight into these changes as well as their manageability in terms of portfolio and project management and decision making. By defining concepts such as work package (to model implementation work at different levels of granularity, e.g., programs, projects, or project tasks), deliverable and plateau it is possible to connect ArchiMate with program and project management standards and best practices, such as MSP [17], PRINCE2 [19] and PMBoK [18]. Deliverables realize plateaus or core elements, and plateaus aggregate core elements. Relationships can be established between the enterprise architecture models created at different
moments in time and the migration models. These differences are captured by the concept of gap. A gap is an important outcome of a gap analysis in Phases B, C, and D of the TOGAF ADM, and forms an important input for the subsequent implementation and migration planning. The gap concept is linked to two plateaus (e.g., baseline and target architecture, or two successive transition architectures), and represents the differences between these plateaus.

**ArchiMate and TOGAF**

Figure 6 shows a global mapping of the ArchiMate framework (core and extensions) to the phases of the TOGAF ADM.

As indicated before, the core concepts are chiefly used to model the actual architectures in Phases B, C and D (“Getting the architecture right”). Moreover, the three layers of the ArchiMate core framework closely correspond to the three main types of architectures that these phases deal with. The concepts of the motivation extension are especially useful in the early ADM phases (“Getting the organization committed and involved”), to support stakeholder analysis and for the formulation of business goals and principles, as well as in Change Management and the Requirements Management phases (“Keep the process running”) to model and manage the requirements and constraints. Finally, the concepts of the implementation & migration extension mainly support the late ADM phases (“Making the architecture work”), to model implementation programs and projects, as well as plateaus and gaps for migration planning.

![Figure 6. TOGAF ADM and ArchiMate (from [6])](image)

**Example case: ArchiSurance**

Using a small example based on a fictitious insurance company, we illustrate how the different architectures as defined in TOGAF can be expressed with concepts of the ArchiMate core. ArchiSurance is a merger of three previously independent companies: Home & Away for homeowner’s and travel insurance, PRO-FIT for car insurance, and Legally Yours for legal aid insurance. The new company has a single Front Office and three separate Back Offices. ArchiSurance has plans to rationalize their application portfolio, by integrating legacy applications with similar functionality from the old companies that are still in use. Note that these examples give an impression of the ArchiMate language, but do not show all the concepts. For a complete overview of the language, please refer to [6].
**Motivation**

Figure 7 shows a fragment of the motivation model for the proposed change. Stakeholder concerns, modelled as drivers, lead to business goals. These goals are realized by principles, which in turn are made more specific in terms of requirements.

**Business Architecture**

The Business Architecture provides the context for system development trajectories, showing, among others, the main business processes, the actors (or roles) performing these processes, and the information (objects) exchanged between the processes.

Figure 8 shows an example of a Business Architecture expressed in ArchiMate. We assume that the business architecture of ArchiSur-ance does not change in the application rationalization process.
Application Architecture

The Application Architecture shows the applications or application components, their relationships and their functionality. Figure 9 shows the baseline Application Architecture of ArchiSurance. The functionality that the applications offer to their environment is modelled with services. The service concept plays a central role in ArchiMate, also in the Business Architecture and the Technology Architecture (although this is not shown in our example), and in particular as a linking pin between the different architectures.

Figure 10 shows the target Application Architecture of ArchiSurance, in which the legacy applications have been replaced by a single back-office system and a single CRM system for the whole company.

Figure 8. Baseline and Target Business Architecture
In ArchiMate, separate views can be used to show the relationships between the different architectures. As an example of this, Figure 11 shows how the services from the Application Architecture are used in the processes of the Business Architecture.

**Figure 11. Business-Application Alignment (Target)**

### Data Architecture

The Data Architecture shows the main data objects used within the applications, as well as their relationships. Figure 12 shows the Data Architecture of ArchiSureance, which we assume will not change in the application rationalization process.

**Figure 12. Baseline and Target Data Architecture**
Technology Architecture

The Technology Architecture shows, among others, the devices and system software on which applications run, the networks connecting devices, and artefacts that form the physical implementation of application components or data objects. Figure 13 shows the baseline Technology Architecture of ArchiSurence. There are separate application servers for the different back-office applications.

In the target Technology Architecture, as shown in Figure 14, some of these application servers become redundant. However, to increase reliability and availability, an additional backup server is introduced.

Gap Analysis

An important step in Phases B, C and D of the TOGAF ADM is a gap analysis, which reviews the differences between the baseline and target architecture. It shows which building blocks are carried over from baseline to target, which building blocks are new in the target architecture (which can be used as a basis to decide whether to buy or build these building blocks), and which elements have been eliminated from the baseline architecture (on purpose or accidentally; i.e., a gap analysis can also be used as a mechanism for validation of the target architecture). Subsequently, Phases E, F and G of the TOGAF ADM deal with the implementation of the proposed target architecture.
TOGAF suggests the use of a gap matrix as a technique for gap analysis. However, ArchiMate models also form a useful starting point for gap analysis, and the results can also be presented as an ArchiMate view. Figure 15 shows an example of this for the Technology Architecture.

Implementation and migration planning

Figure 16 shows an example of the use of work packages and deliverables. It shows one of the implementation projects, the integration of the back-office systems of ArchiSure. This project is broken down in a number of project tasks, that product well-defined deliverables.

An important premise in TOGAF is that the various architectures are described for different stages in time. In each of the Phases B, C, and D of the ADM, a Baseline Architecture and Target Architecture are created, describing the current situation and the desired future situation. In Phase E, Transition Architectures are defined, showing the enterprise at incremental states reflecting periods of transition between the Baseline and Target Architectures. Transition Architectures are used to allow for individual work packages and projects to be grouped into managed portfolios and programs, illustrating the business value at each stage. In order to support this, the plateau concept was introduced. Figure 17 shows an example with a single intermediary transition architecture. For example, after completion of the back-office integration project as described above, an intermediary situation is created with a single integrated back-office, but...
still two separate CRM applications. The gap concept is used to make the differences between two plateaus explicit (a gap is generally associated with elements in the architecture that are modified between the plateaus).

Summary and Conclusions

For several years, TOGAF has been the leading enterprise architecture method, developed and maintained by members of The Open Group. More recently, ArchiMate has been adopted as another standard of The Open Group, aimed at modelling enterprise architectures. With the publication of Version 2.0 of ArchiMate in early 2012, adding two extensions to the core language, thus providing complete modelling support throughout the architecture development cycle. TOGAF and ArchiMate have a shared vision on enterprise architecture the use of viewpoints, and the concept of an underlying common repository of architectural artefacts and models; i.e., they have a firm common foundation. However, they complement each other with respect to the definition of an architecture development process and the definition of an enterprise architecture modelling language. ArchiMate provides a concrete visualization for the architectures and views proposed in TOGAF. Thus, these two complementary open standards will reinforce each other and help to advance the enterprise architecture discipline in general.
References


Please contact us at info@bizzdesign.com, or send a direct message to Henk Jonkers, h.jonkers@bizzdesign.com

BiZZdesign Offices

Corporate Headquarters
R & D and Backoffice
Capitool 15
P.O.Box 321
7500 AH Enschede
The Netherlands
+31 (0)53 4 878 151
info@bizzdesign.com

BiZZdesign Netherlands
Consultancy and Academy
 Stationsstraat 79 F (6th floor)
3811 MH Amersfoort
The Netherlands
+31 (0)33 4 613 754
sales@bizzdesign.com

BiZZdesign North America
161 Bay Street, 27th Floor
Toronto, ON M5J 2S1
Canada
+416.572.2035
info@bizzdesign.com

BiZZdesign United Kingdom
16th Floor
Portland House
Bressenden Place
London
SW1E 5RS
United Kingdom
+44 (0) 207 869 8072
info@bizzdesign.com

BiZZdesign Deutschland GmbH
Lortzingstrasse 12
40667 Meerbusch-Büderich
Germany
+49 2132 692 0387
info@bizzdesign.com

BiZZdesign Belgium
Geldenaksebaan 48
3001 Leuven
Belgium
+32 497 302387
info@bizzdesign.com