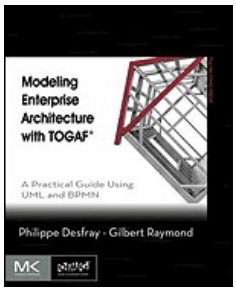

Modeling Enterprise Architecture with TOGAF: A Practical Guide Using UML and BPMN

by Philippe Desfray and Gilbert Raymond

Chapter 6: Introduction to TOGAF Models

TOGAF defines a large number of different kinds of artifacts. This chapter provides examples of artifacts and interpretations of how they can be realized and modeled. This will help you to get to know TOGAF artifacts and how to use them and will provide useful modeling techniques to realize them. The progress of artifact construction is illustrated through an example of an enterprise.

6.1 TOGAF ARTIFACTS

6.1.1 Using Models to Realize Artifacts

The notion of the TOGAF *artifact* was described in Section 3.3. Artifacts are a means of communication that present a particular view of the architecture. Catalogs, matrices, and diagrams are three types of artifacts. Architecture objects (for example, actor, business entity, business process) will be represented in these artifacts, along with their properties and links. TOGAF provides a list of recommended artifacts by architecture domain and also indicates which ADM phase uses or produces which artifacts.

Based on the list of artifacts recommended by TOGAF, this book describes how these can be represented using the Unified Modeling Language (UML) and Business Process Modeling Notation (BPMN) modeling standards or specific extensions.

> **Note** The model examples provided in this chapter and in Chapters 7–11 can be downloaded from www.togaf-modeling.org/downloads-menu.html and used with the open source Modelio tool.[1]

Of course, in the first instance, models support "diagram"-type artifacts. However, matrices and catalogs can also be produced (generated) from models too. Consequently, we sometimes propose diagram-type artifacts to support certain TOGAF catalogs.

It should be noted that TOGAF does not describe how to model artifacts in detail. The implementation of the models in this book is therefore a specific creation on the part of the authors and is the result of their choices.

6.1.2 Preliminary Phase: Determining Useful Artifacts in the Context of the Enterprise

Let's remember that TOGAF is a generic methodological framework. Every enterprise and every context will require that TOGAF be adapted. This adaptation takes place during the *preliminary phase*, in the context of the "Tailored Architecture Framework" deliverable and, more particularly for artifacts, the "Tailored Architecture Content" section. For this reason, TOGAF proposes a list of artifacts but does not impose that they all be realized, and does not claim that the list is exhaustive. This book has, therefore, partially carried out this adaptation work in order to facilitate the support of TOGAF by modeling standards. It has made a selection, which includes a large majority of TOGAF artifacts, as well as other artifacts that are considered useful for enterprise architecture. Readers can use this work as a basis for the completion of this adaptation to their own context.

The aim of the preliminary phase (see Section 2.2.1) initiating the TOGAF ADM is precisely to determine the viewpoints and artifacts that are considered to be important in the context of an enterprise. Stakeholders participating in enterprise architecture work must therefore be identified. Once we have identified these actors' specific issues, we can then determine the necessary representation viewpoints. By default, TOGAF architecture domains can be used (this is the choice made in this book) or specific viewpoints can be defined.

From a pragmatic standpoint, we recommend that participants in enterprise architecture work be identified (as TOGAF actors), that artifacts be reviewed and only those relevant to the enterprise be retained, and that their characteristics be redefined, notably by specifying the nature of the participants involved in each artifact for the enterprise.

To facilitate this selection, we have characterized each artifact using the following properties:

- Name
- Participants: The aim of the preliminary phase is to identify the enterprise stakeholders who are involved in enterprise architecture activities. For each artifact, we must indicate which of these participants contributes to its elaboration. Among participants, the following categories[2] can be distinguished:
  - Experts: Those who bring knowledge. For example, the executive management of the company is responsible for determining goals.
  - Designers: Those who realize the artifact. For example, business analysts can elaborate the goal model.
  - Recipients: Those for whom the artifact is destined. For example, the definition of goals is essentially destined for business analysts and architects, who must refer to them. These participants are generally involved in the review of the current artifact.
- Aim: What is the benefit and usefulness of the artifact? To do what? How is it used? This information is decisive when deciding whether or not the artifact should be included in the enterprise architecture work of a given enterprise (benefits versus cost of construction).
- Incoming elements: List of information extracted from other artifacts and useful in the development of the current artifact.

6.1.3 Structuring Artifacts

The models presented in this book are based on the artifacts provided by TOGAF, as shown in Figure 6.1. In this figure, artifacts are organized either by type of architecture (for example, data architecture) or by phase (for example, Phase E—opportunities and solutions).

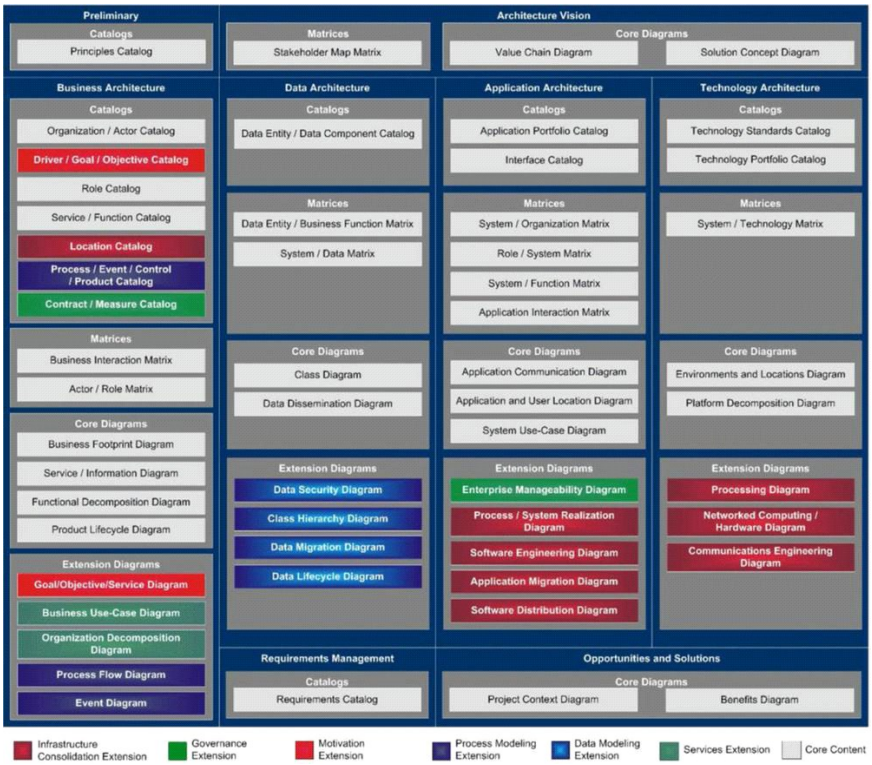

Figure 6.1: Different artifacts defined by TOGAF (extract from the reference document)

Generally speaking, several phases contribute to the development and consolidation of an artifact. Thus, certain artifacts may be initiated during the vision phase, which presents the initial architectural intentions in a very general manner, before being consolidated, notably during phases B and C. This book structures these artifacts according to the phase they are most involved in.

The chapters dealing with artifacts are organized as follows:

- Chapter 7—Vision (essential artifacts of phase A)
- Chapter 8—Phase B (business architecture)
- Chapter 9—Phase C (application architecture)
- Chapter 10—Phase D (technology architecture)
- Chapter 11—Phase E (opportunities and solutions)

In Section Organization of the model, we see that data architecture will be split into two parts, one related to business architecture and the other to data architecture.

Quite logically, these chapters focus more on the development phases (notably A, B, C, and D), which are more concerned with the development of artifacts linked to models (Table 6.1).

Table 6.1: Examples of "Diagram" Artifacts Organized by Phase

Artifact	Main Phase	Description
Solution concept diagram	A: Architecture vision	High-level orientation of the envisaged solution (its main components) to reach the goals of the enterprise architecture
Organization/actor catalog	B: Business architecture	Definition of the actors, their duties, hierarchical links, and responsibilities
Functional decomposition diagram	B: Business architecture	Graphical representation of the functions of the enterprise, structured hierarchically
Process flow diagram	B: Business architecture	Detailed view of the functioning of a process, broken down into tasks
Class diagram (business level)	B: Business architecture	Modeling of business entities and their properties and associations
Application communication diagram	C: Application architecture	Representation of interconnections and communication between the system's applications and application components
Application migration diagram	C: Application architecture	Representation of the IS's evolution path, with its different stages

Networked computing diagram	D: Technology architecture	Representation of the hardware and network architecture (servers, networks, etc.)
Requirements analysis diagram; requirements catalog	Requirements management	Definition of requirements and their properties and links
Benefits diagram	E: Opportunities and solutions	Representation of opportunities and solutions at application architecture level

6.1.4 Organization of the Model

It is a common practice to organize TOGAF enterprise architecture model using a structure that resembles the viewpoints retained for the approach (see Section 5.2.4) as closely as possible. Since UML and BMPN modeling tools do not use the concept of viewpoints, the model must be organized using *packages*,[3] which represent these viewpoints. By default, viewpoints span at least the four TOGAF architecture domains: Business, Application, Data, and Technology. The separation of concepts and models is easy with regard to business, application, and technology architectures and is naturally supported by UML packages that are specialized for TOGAF.

Data architecture partially spans business architecture and application architecture, depending on the level of representation used. We will see examples of these two data architectures at the business level in Chapter 8 and at the application architecture level in Chapter 9.

Two subpackages are therefore created inside the business architecture and application architecture packages to support data architecture.

This structuring is by no means mandatory (Figure 6.2). In particular, the definition of viewpoints specific to an enterprise can lead to different structuring. For example, an additional viewpoint dedicated to system security could be envisaged.

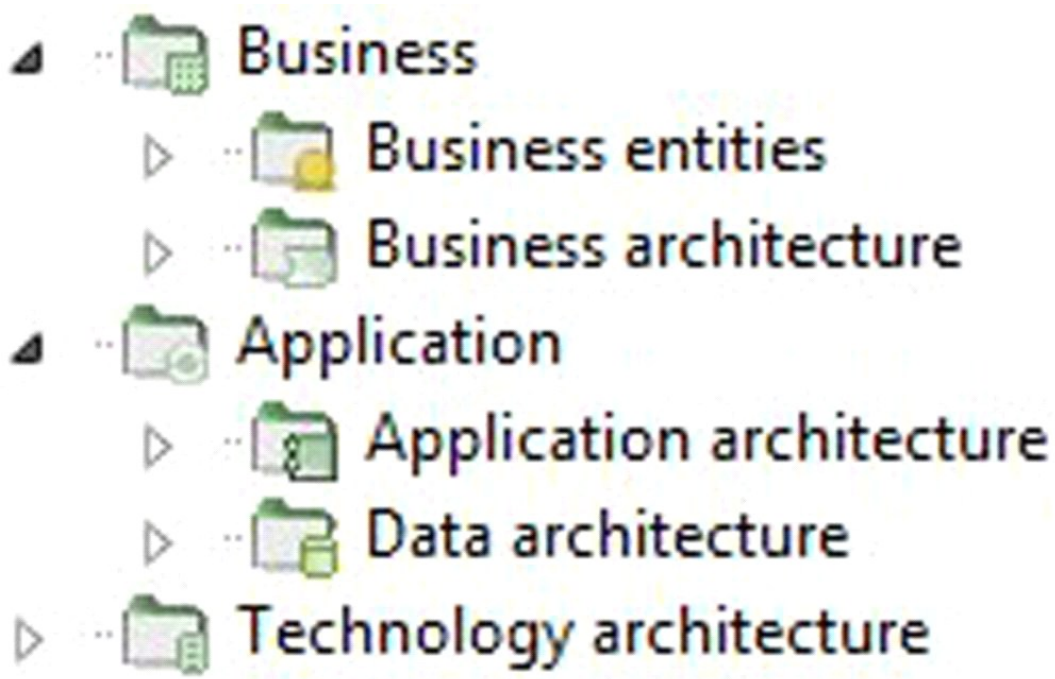

Figure 6.2: Structuring of different TOGAF architecture into packages

The structure is then further broken down in a functional way, relevant to the enterprise's business.

[1]www.modelio.org.

[2]These categories are usually formalized through a RACI (responsibility assignment matrix showing the "responsible, accountable, consulted, informed" roles per deliverable).

[3]UML concept enabling model elements to be structured.

6.2 UML AND BPMN FOR TOGAF MODELING

6.2.1 Choosing a Representation Mode for TOGAF Models

TOGAF refers several times to the UML and BPMN modeling languages to support enterprise architecture modeling. However, some explanations are still required on how these standards should be used to apply TOGAF, and on which model parts to show for each viewpoint.

UML and BPMN will be used to represent basic architectural objects, defined in the TOGAF metamodel (see Section 3.2.1). This means that we must therefore decide which UML or BPMN element will be used to represent each TOGAF object. In this way, a BPMN process will naturally represent a TOGAF process, a UML actor will represent a TOGAF actor but also a TOGAF role, and a UML class will represent a TOGAF business entity but also a TOGAF product.

As we saw in Chapter 5, models must be adapted to facilitate communication between participants. UML and BPMN, both extremely rich standards, must be filtered according to viewpoints so as only to present useful concepts and must also be adapted to correspond to TOGAF terminology (such as a role, a business entity, etc.). They sometimes have to be extended to support concepts that are absent from these languages (for example, a function, an organization unit, a goal).

6.2.2 Modeling Standards for TOGAF

Panorama of Standards That are Useful for TOGAF

Most modeling standards used in this book stem from the OMG organization. The OMG has standardized very well-known standards such as UML and BPMN and has workgroups that provide standards for modeling domains linked to enterprise architecture (vision, enterprise organization, process modeling, requirements modeling, SOA architecture modeling). It is therefore a very rich and useful source for selecting modeling techniques adapted to TOGAF.

- UML and BPMN are essential standards defined by the OMG and used in enterprise architecture modeling.
- Service component architecture is a reference in the field of SOA architecture definition. SoaML is an OMG standard based on UML and used in SOA architecture modeling.
- SysML is an OMG standard dedicated to modeling technical systems. It models systems by breaking them down into components, fitting together systems, subsystems, and components (the concept of "blocks"). It provides a requirements modeling standard reused in this book.
- Business motivation metamodel is an OMG standard providing a detailed metamodel for the "who, what, why, and how" of business motivation (vision phase, goal definition).
- Organizational structure metamodel is an OMG specification (not an adopted standard) defining useful concepts for organization modeling. Here, we find concepts similar to those of TOGAF, such as the concept of the organization unit.
- Ontology definition metamodel is an OMG standard, providing a metamodel for the definition of ontologies. Part of this metamodel is used to support the concept of the dictionary.

The UML Standard

UML was standardized in 1997, and a major new version was published in 2005. UML groups together a large number of modeling techniques that were previously scattered among different domains (entity relationship, object model, state diagram, sequence diagram, process modeling, etc.). It is widely accepted and used in the modeling of software systems.

UML enables data to be modeled through class diagrams. Behavior is modeled through object modeling (object behaviors, operations, etc.) and the support of sequence diagrams, state diagrams, and activity diagrams. Systems and architectures are also modeled using the concept of components and component assembly techniques.

Extending and Adapting UML: The Profile Mechanism

UML is used for a wide variety of domains and targets and must therefore be adapted to correspond to the concerns and concepts specific to each target. For example, the concept of persistence must be introduced when modeling database schemas, while concepts specific to programming languages must be added for targets. Similarly, the concept of service is required when modeling SOA architectures, while the concept of the system is central to large system modeling.

A UML *profile* is a set of extensions brought to UML to adapt it to a particular target. For example, SoaML or SysML (Figure 6.3) are UML profiles standardized by the OMG.

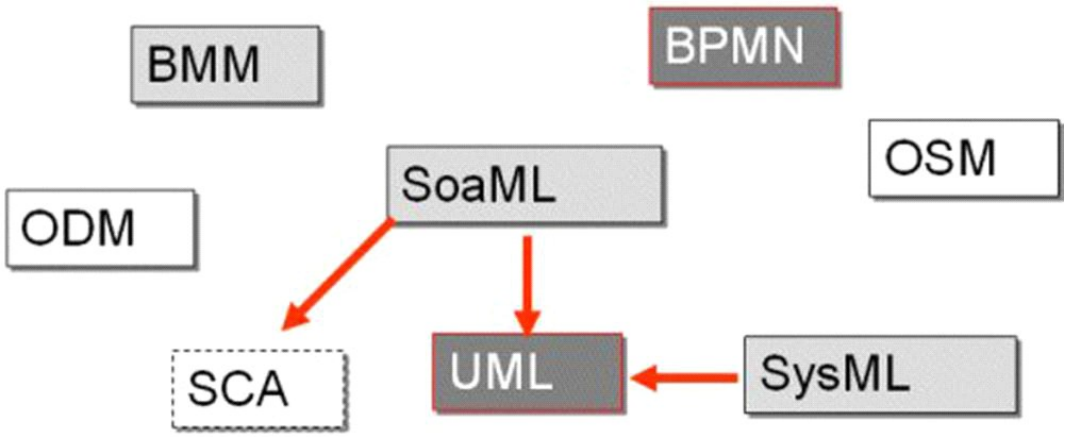

Figure 6.3: Reference standards for supporting enterprise architecture modeling

We will see that in order to better target TOGAF, a UML profile has also been defined in this book.

The BPMN Standard

BPMN is a standardized, graphical notation used to model business processes and workflows.

The main goal of BPMN is to provide notation that is truly understandable by all enterprise users, from business analysts who create initial sketches of processes, through developers in charge of setting up the technology that will run these processes, right up to enterprise users who will manage and supervise these processes.

BPMN was standardized by the OMG in 2006, and a major new version (BPMN2) published in 2010.

Although the UML standard also has notation for modeling processes (activity diagrams), BPMN is an OMG standard that is independent of UML. In practice, several modeling tools support both modeling languages.

Adapting Standards to TOGAF: The Enterprise Architecture Profile

TOGAF provides a metamodel that describes the key concepts (TOGAF objects) that have to be linked to the concepts provided by UML and BPMN. This is why this book provides a UML profile dedicated to TOGAF.[4] This "UML *profile*," named EAP (*Enterprise Architecture Profile*) extends the relevant UML concepts in order to represent all TOGAF objects. For example, a UML class is the best adapted concept to represent a TOGAF "business entity" (Phase B), but also a "message," or "data" in phase C. Figure 6.4 shows how these different concepts, all of which are based on the UML class concept, are distinguished by particular extensions. Figure 6.5 shows the representation of other TOGAF objects supported by the EAP.

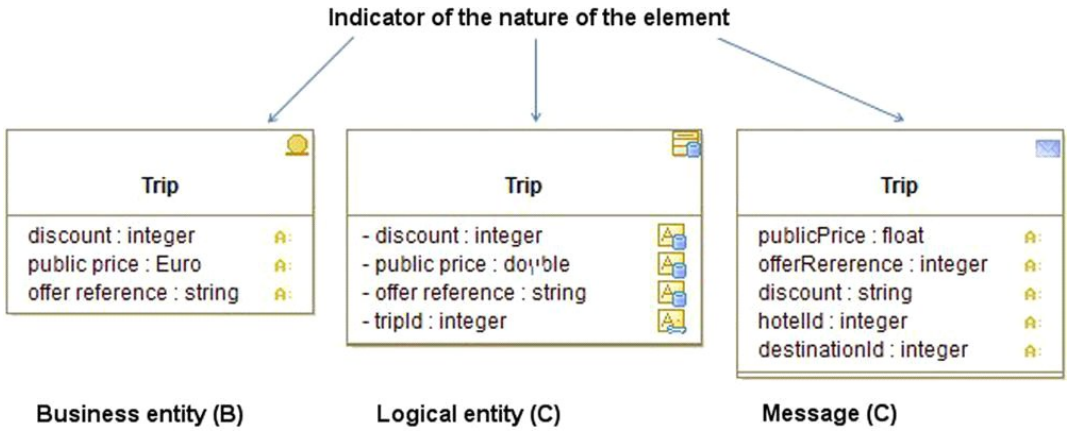

Figure 6.4: The UML profile for TOGAF distinguishes TOGAF concepts

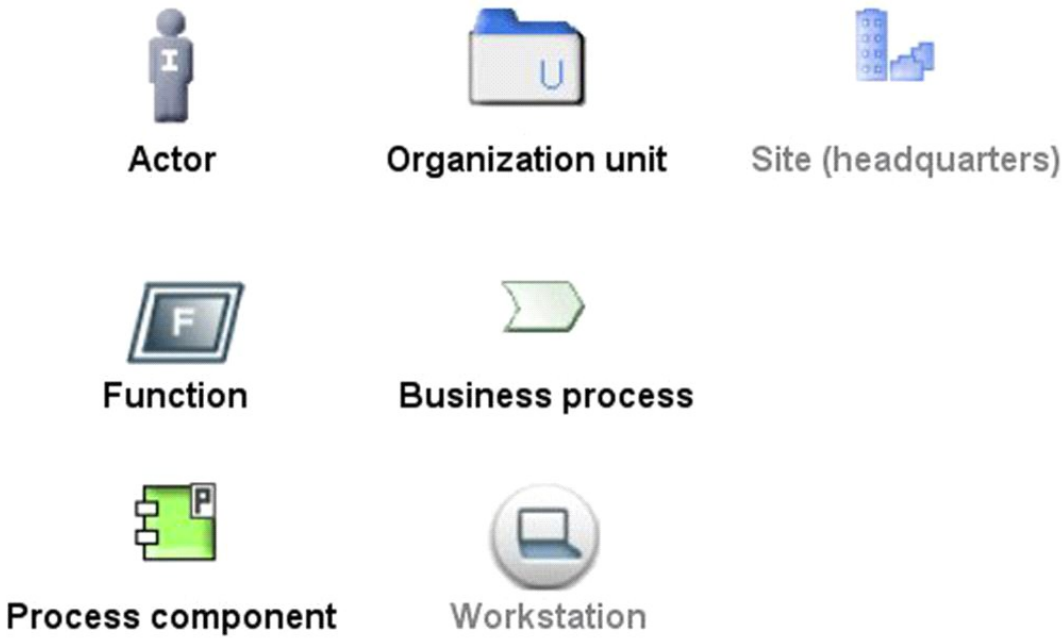

Figure 6.5: TOGAF objects represented using the EAP

The use of UML with the EAP and of BPMN therefore enables us to reuse the standards shown in Figure 6.3 by bringing them together in a language dedicated to TOGAF. The open source Modelio tool is used to represent the TOGAF models presented in this book. However, since UML, BPMN, and profile implementation are standard techniques, other modeling tools available in the marketplace can also be used.

This also has the advantage of reusing modeling techniques referred to by TOGAF and are explicitly covered by these standards, such as Use Cases (UML) or BPMN.

[4]The EAP profile can be freely downloaded from www.togaf-modeling.org/togaf-en-pratique/. Documentation on this profile can also be found here.

6.3 THE "DISCOUNT TRAVEL" ENTERPRISE

Throughout our entire presentation of artifacts, we use the same enterprise example, that of the "Discount Travel" enterprise. We based this example on specifications provided by Ceisar.[5]

Discount Travel is a service provider that provides the public with a list of trips that have not been sold by travel agencies.

Different trip/holiday formats are proposed at reduced prices. Discounts can reach 50% of the published price applied by travel agencies.

The prices offered by Discount Travel are explained by the fact that the departure and return dates are fixed, and are often imminent. The future client must therefore be willing to depart within 2 weeks of making his or her reservation.

Until now, Discount Travel has provided a telephone service open to the public from 8 am to 8 pm from Monday to Friday. The potential client selects the trip that is of interest to him or her with the help of a customer advisor. The processes of the enterprise are not formalized and the information used by customer advisers to respond to clients takes the form of paper documents, updated daily by the marketing department according to the evolution of the stock of available products.

A website exists that currently enables order elements to be entered. These order elements are then processed manually by an agent. Orders are recorded using an application accessed by sales representatives.

Discount Travel is considering the possibility of providing a reservation service online. Moreover, the enterprise wishes to improve its customer service. The design of the information system must therefore be reviewed.

A trip corresponds to a format (combination of flight, hotel, and car rental), a destination, and an accommodation service.

Destinations are identified by the continent (North Africa, Africa (except North Africa), Europe, Asia, and America) and the country.

The trip takes place in one country only.

The relationship with travel agencies is managed by the marketing department, who defines priority products to search for among the range of trips available in travel agencies. The goal of the marketing department is to develop the most attractive range of trips possible for the clients of Discount Travel.

In our work on this example, we go into detail on the organization of the enterprise, in particular clarifying the enterprise's goals and business processes. We also describe business entities (what a trip is, a client file, etc.), geographical locations, the enterprise's business and IS

requirements, application architectures, and technology architectures.

[5]http://www.ceisar.fr/: Center of excellence in enterprise architecture, affiliated with the Ecole Centrale in Paris.

6.4 FUNDAMENTAL CONCEPTS

The following fundamental concepts were introduced in this chapter:

- EAP: Profile that adapts and filters UML in conjunction with BPMN to map it to TOGAF enterprise architecture modeling.