

The Open Group Guide

**ArchiMate® Modeling Notation for the
Financial Industry Reference Model:
Banking Industry Architecture Network (BIAN)**



Copyright © 2020, The Open Group and BIAN Association

The Open Group hereby authorizes you to use this document for any purpose, PROVIDED THAT any copy of this document, or any part thereof, which you make shall retain all copyright and other proprietary notices contained herein.

This document may contain other proprietary notices and copyright information.

Nothing contained herein shall be construed as conferring by implication, estoppel, or otherwise any license or right under any patent or trademark of The Open Group or any third party. Except as expressly provided above, nothing contained herein shall be construed as conferring any license or right under any copyright of The Open Group.

Note that any product, process, or technology in this document may be the subject of other intellectual property rights reserved by The Open Group, and may not be licensed hereunder.

This document is provided “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. Some jurisdictions do not allow the exclusion of implied warranties, so the above exclusion may not apply to you.

Any publication of The Open Group may include technical inaccuracies or typographical errors. Changes may be periodically made to these publications; these changes will be incorporated in new editions of these publications. The Open Group may make improvements and/or changes in the products and/or the programs described in these publications at any time without notice.

Should any viewer of this document respond with information including feedback data, such as questions, comments, suggestions, or the like regarding the content of this document, such information shall be deemed to be non-confidential and The Open Group shall have no obligation of any kind with respect to such information and shall be free to reproduce, use, disclose, and distribute the information to others without limitation. Further, The Open Group shall be free to use any ideas, concepts, know-how, or techniques contained in such information for any purpose whatsoever including but not limited to developing, manufacturing, and marketing products incorporating such information.

If you did not obtain this copy through The Open Group, it may not be the latest version. For your convenience, the latest version of this publication may be downloaded at www.opengroup.org/library.

The Open Group Guide

ArchiMate® Modeling Notation for the Financial Industry Reference Model: Banking Industry Architecture Network (BIAN)

ISBN: 1-947754-52-2

Document Number: G205

Published by The Open Group, March 2020.

Comments relating to the material contained in this document may be submitted to:

The Open Group, Apex Plaza, Forbury Road, Reading, Berkshire, RG1 1AX, United Kingdom

or by electronic mail to:

ogspecs@opengroup.org

Contents

1	Introduction.....	1
1.1	Objective.....	1
1.2	Scope.....	1
1.3	Overview.....	1
2	The Banking Industry Architecture Network (BIAN)	3
3	BIAN Service Landscape, Version 7.0	5
3.1	Business Areas.....	6
3.2	Business Domains.....	7
3.3	Service Domains.....	8
3.3.1	Action Terms.....	10
3.3.2	Functional Pattern.....	11
3.3.3	Generic Artifact.....	12
3.4	Service Operations.....	14
3.4.1	Input/Output Message	15
3.4.2	Pre/Post-Condition	15
4	BIAN Business Scenarios and Wireframes.....	17
4.1	Business Scenarios.....	17
4.2	Wireframes	18
5	BIAN Business Object Model Framework	20
5.1	BIAN Business Object Model	20
5.2	Business Object Modeling: The Approach	21
5.2.1	Business Object <i>versus</i> Business Concept	21
5.2.2	Content Pattern.....	22
5.2.3	Structure Pattern.....	24
5.2.4	Classification Mechanism	25
5.2.5	Example Payments	26
6	Case Study: Archi Banking Group.....	28

Preface

The Open Group

The Open Group is a global consortium that enables the achievement of business objectives through technology standards. Our diverse membership of more than 700 organizations includes customers, systems and solutions suppliers, tools vendors, integrators, academics, and consultants across multiple industries.

The mission of The Open Group is to drive the creation of Boundaryless Information Flow™ achieved by:

- Working with customers to capture, understand, and address current and emerging requirements, establish policies, and share best practices
- Working with suppliers, consortia, and standards bodies to develop consensus and facilitate interoperability, to evolve and integrate specifications and open source technologies
- Offering a comprehensive set of services to enhance the operational efficiency of consortia
- Developing and operating the industry's premier certification service and encouraging procurement of certified products

Further information on The Open Group is available at www.opengroup.org.

The Open Group publishes a wide range of technical documentation, most of which is focused on development of Standards and Guides, but which also includes white papers, technical studies, certification and testing documentation, and business titles. Full details and a catalog are available at www.opengroup.org/library.

The Banking Industry Architecture Network (BIAN)

The Banking Industry Architecture Network (BIAN) is a collaborative not-for-profit ecosystem formed of leading banks, technology providers, consultants, and academics from all over the globe.

BIAN is created to establish, promote, and provide a common framework for banking interoperability issues and to become and to be recognized as a world-class reference point for interoperability in the banking industry.

For more information refer to www.bian.org.

This Document

This document is The Open Group Guide to ArchiMate® Modeling Notation for the Financial Industry Reference Model: Banking Industry Architecture Network (BIAN). It has been developed and approved by The Open Group and BIAN.

This document is structured as follows:

- Chapter 1 provides an overview of how the ArchiMate Specification, a standard of The Open Group, can be used to exploit the value of the BIAN Financial Industry Reference Model (BIAN Reference Model) in managing the transition to a digital financial institution
- Chapter 2 introduces the BIAN organization and outlines the intended benefits to the banking industry of combining its BIAN Reference Model with the ArchiMate language
- Chapter 3 introduces the BIAN metamodel as expressed in the ArchiMate 3.0.1 language, explaining how the metamodel supports the construction of the BIAN Reference Model
- Chapter 4 introduces the business scenarios and wireframes of the BIAN, which are process-oriented views on its service landscape
- Chapter 5 describes the Business Object Modeling (BOM) approach, used to enrich the BIAN Reference Model with a reference architecture model for the financial industry
- Chapter 6 presents the Archibank case study, developed to illustrate the use of the BIAN Reference Model

Trademarks

ArchiMate®, DirecNet®, Making Standards Work®, Open O® logo, Open O and Check® Certification logo, OpenPegasus®, Platform 3.0®, The Open Group®, TOGAF®, UNIX®, UNIXWARE®, and the Open Brand X® logo are registered trademarks and Agile Architecture Framework™, Boundaryless Information Flow™, Build with Integrity Buy with Confidence™, Dependability Through Assuredness™, Digital Practitioner Body of Knowledge™, DPBoK™, EMMM™, FACE™, the FACE™ logo, FBPT™, FHIM Profile Builder™, the FHIM logo, IT4IT™, the IT4IT™ logo, O-AAF™, O-DEF™, O-HERA™, O-PAS™, Open FAIR™, Open Platform 3.0™, Open Process Automation™, Open Subsurface Data Universe™, Open Trusted Technology Provider™, O-SDU™, OSDU™, Sensor Integration Simplified™, SOSA™, and the SOSA™ logo are trademarks of The Open Group.

IBM® is a registered trademark of International Business Machines Corporation.

ITIL® is a registered trademark of AXELOS Limited.

Microsoft® is a registered trademark of Microsoft Corporation.

SAP® is a registered trademark of SAP SE in Germany and/or other countries.

SBVR™, Semantics of Business Vocabulary and Business Rules™, and Unified Modeling Language™ are trademarks, and UML® is a registered trademark of the Object Management Group, Inc. in the United States and/or other countries.

SWIFT™ is a registered trademark in Belgium and/or other countries.

All other brands, company, and product names are used for identification purposes only and may be trademarks that are the sole property of their respective owners.

Acknowledgements

The Open Group gratefully acknowledges the contribution of the following people in the development of this Guide:

Work Group members:

- Patrick Derde (Chair), EnVizion, BIAN
- Sonia Gonzalez, The Open Group
- Marc Lankhorst, BiZZdesign

Co-authors:

- Martine Alaerts, EnVizion
- Laleh Rafati, EnVizion

Referenced Documents

The following documents are referenced in this Guide.

(Please note that the links below are good at the time of writing but cannot be guaranteed for the future.)

- A Complete and Consistent Business: Introduction to the COSTA Model for Business Architects, K. Knaepen, D. Brooms, Lannoo Publishers, 2013
- Analytics: The Speed Advantage – Why Data-driven Organizations are Winning the Race in Today’s Marketplace, IBM® Institute for Business Value, 2014; refer to: www.ibm.com/thought-leadership/institute-business-value/report/2014analytics
- ArchiMate® 3.1 Specification, a standard of The Open Group (C197), November 2019, published by The Open Group; refer to: www.opengroup.org/library/c197
- ArchiMate® Model Exchange File Format for the ArchiMate Modeling Language, Version 3.1, a standard of The Open Group (C19C), November 2019, published by The Open Group; refer to: www.opengroup.org/library/c19c
- BIAN “How-to” Guide; refer to: www.bian.org/deliverables/bian-how-to-guide
- BIAN Metamodel Specification, Version 7.0, BIAN Architectural Framework and Foundation Working Group, Version: 7.0, 2018; refer to: www.bian.org
- Case Study: Archi Banking Group (Y201), March 2020, published by The Open Group; refer to: www.opengroup.org/library/y201
- ISO 20022: Financial Services – Universal Financial Industry Message Scheme; refer to: www.iso20022.org/the_iso20022_standard.page
- The TOGAF® Standard, Version 9.2, a standard of The Open Group (C182), April 2018, published by The Open Group; refer to: www.opengroup.org/library/c182

The following documents provide useful background information:

- A Case for Enterprise Data Management in Banking, Capgemini, 2012; refer to: www.capgemini.com/resources/a-case-for-enterprise-data-management-for-banking/
- Banking Industry Architecture Network (BIAN) – Standardization for Semantic Interoperability, D. Frankel, MDA Journal, 2011; refer to: www.bptrends.com/mda-journal-banking-industry-architecture-network-bian-standardization-for-semantic-interoperability/
- BIAN Edition 2019: A Framework for the Financial Services Industry, The BIAN Association, Van Haren Publishing, 2018
- Semantics of Business Vocabulary and Business Rules™ (SBVR™), Version 1.4, Object Management Group, 2017; refer to: www.omg.org/spec/SBVR/1.4/PDF

1 Introduction

1.1 Objective

This document provides guidance on how the ArchiMate® Specification, a standard of The Open Group, can be used to exploit the value of the Banking Industry Architecture Network (BIAN) Financial Industry Reference Model (hereafter referred to as the “BIAN Reference Model”).

It is designed to provide a guide to Business Managers, Business Architects, Business Change Managers, CIOs, Enterprise Architects, Solution Architects, IT executives, IT professionals, and all individuals involved or interested in how to manage the transition to a digital financial institution.

It guides an Enterprise Architecture organization to develop an agile, lean, and stable banking architecture using the ArchiMate language and BIAN.

1.2 Scope

This document is based on the BIAN Metamodel Specification, Version 7.0, and is accompanied by examples from the BIAN Service Landscape, Version 7.0. This document contains high-level explanatory material. For further details see the BIAN “How-to” Guide.

The BIAN metamodel is currently expressed in Unified Modeling Language™ (UML®) class diagrams as an extension of the ISO 20022 metamodel. This document does not contain the specification of the ISO 20022 metamodel, which is available from ISO (see [Referenced Documents](#)).

This document expresses the BIAN metamodel in ArchiMate diagrams.

1.3 Overview

After many years of developing and improving the BIAN Reference Model, applying many best practices (and standards), configuring Enterprise Architecture modeling tools, creating BIAN-specific workbenches, and explaining the meaning of BIAN concepts, BIAN architects admitted that the metamodel concepts are not commonly understood and are hard to explain to the banking architects using the BIAN Reference Model. The BIAN Enterprise Architect team decided to express the BIAN concepts in the ArchiMate language, making use of the extension mechanisms of the ArchiMate Specification; i.e., profiles and specializations. In this document only specialization is used. The ArchiMate concepts are specialized whenever BIAN uses a concept that is a specialization of a basic ArchiMate concept. This allows the BIAN content to be much more easily explained by using a common language between Enterprise Architects.

The BIAN Reference Model, expressed in the ArchiMate language, and the ArchiMate language itself is a powerful combination for supporting strategic management in the transformation to a digitalized bank, providing a means of using the BIAN Reference Model with the full set of

ArchiMate capabilities. This allows the exchange of BIAN content using tools that support the ArchiMate Model Exchange File Format. Financial institutions, Fintechs¹ and all types of suppliers can exchange BIAN information and information services by using the common ArchiMate modeling language.

Organizing the financial industry operating model based on the BIAN Reference Model combined with the ArchiMate language allows organizations to:

- Focus on the financial services landscape and capabilities that make the company more competitive and innovative
- Extend the BIAN Reference Model with all the related architecture elements such as:
 - Active and passive structure elements and behavioral elements available in the ArchiMate language at the different layers: Strategic, Business, Application, Technology, and Physical
 - The motivational elements at all layers
 - The implementation elements
- Improve the overall performance and efficiency of the use of the BIAN Reference Model to exceed expectations
- Adopt the BIAN Reference Model with the ArchiMate capabilities to enable a financial institution to create a more mature and professional organization-specific banking architecture

This is realized by implementing a standards-based financial reference architecture allowing an integrated and consistent approach for strategic management, integrating tools from different vendors, supporting (and automating) end-to-end workflows, and providing standard interfaces to collaborate with external service providers while leveraging established best practices.

This document first introduces the BIAN Reference Model from a user perspective.

It then explains the concepts used in BIAN from an architect's perspective, using the ArchiMate representation.

¹ See https://en.wikipedia.org/wiki/Financial_technology

2 The Banking Industry Architecture Network (BIAN)

BIAN is a not-for-profit organization with over 70 members worldwide. It was founded in 2008 by (amongst others): SAP®, Microsoft®, ING, IBM®, and Credit Suisse.

BIAN is an association of banks, solution providers, and educational institutions with the shared aim of defining a semantic service operation standard for the banking industry. The expectation of BIAN is that a standard definition of the business functions and service interactions that describe the general internal workings of any bank will be of significant benefit to the industry. When compared to a proliferation of proprietary designs, such an industry standard provides the following main benefits:

- It enables the more efficient and effective development and integration of software solutions for banks
- It improves the operational efficiency within banks and provides the opportunity for greater solution and capability re-use within and among banks
- It supports the adoption of more flexible business service sourcing models, and enhances the evolution and adoption of shared third-party business services

BIAN refers to the collection of designs that make up its industry standard as the BIAN Service Landscape. The development of the BIAN Service Landscape is iterative, relying on the active contribution of industry participants to build consensus and encourage adoption. BIAN coordinates the evolution of the BIAN Service Landscape on behalf of its membership with regular version releases to the industry, and seeks feedback to help continually expand and refine its content.

In the architecture continuum, the BIAN Reference Model aspires to provide the industry architecture for the financial industry. It aspires to provide a common architecture language within a financial institution and between financial institutions, their partners, and suppliers.

Years of effort have been invested in developing and improving the BIAN Reference Model. The BIAN Reference Model and its metamodel have been documented in a BIAN-specific notation and workbench.

In order to make the knowledge transfer to banking architects using the BIAN Reference Model easier, the BIAN Enterprise Architect team decided to switch to the ArchiMate language. This allows the BIAN content to be much more easily explained by using a common language between Enterprise Architects.

The BIAN concepts are expressed making use of the extension mechanisms of the ArchiMate Specification; i.e., profiles and specializations. In this document only specialization is explained. The ArchiMate concepts are specialized whenever BIAN is using a concept that is a specialization of a basic ArchiMate concept.

The BIAN Reference Model, expressed in the ArchiMate language, and the ArchiMate language itself is a powerful combination for supporting strategic management in the transformation to a digitalized bank, providing a means of using the BIAN Reference Model with the full set of

ArchiMate capabilities. This allows the exchange of BIAN content using tools that support the ArchiMate Model Exchange File Format. Financial institutions, Fintechs, and all types of partners and suppliers can exchange BIAN information and information services by using the common ArchiMate modeling language.

Developing an organization-specific operating model based on the BIAN Reference Model allows organizations to focus on the financial services landscape and capabilities that make the company more competitive and innovative.

Using the ArchiMate language enables organizations to extend the BIAN Reference Model with all its related organization-specific architecture elements, such as:

- The core architecture elements (active and passive structure elements, and behavioral elements) available in the ArchiMate Specification at the different layers: Strategic, Business, Application, Technology, and Physical
- The motivational elements at all layers
- The implementation elements

Adoption of the BIAN Reference Model in combination with the capabilities of the ArchiMate language enables a financial institution to create a more mature and professional organization-specific banking architecture.

This is realized by implementing a standards-based financial reference architecture allowing an integrated and consistent approach for strategic management, integrating tools from different vendors, supporting (and automating) end-to-end workflows, and providing standard interfaces to collaborate with external service providers while leveraging established best practices.

3 BIAN Service Landscape, Version 7.0

The metamodel of the BIAN Reference Model modeled in the ArchiMate language is visualized in Figure 1 and will be explained further later in this document.

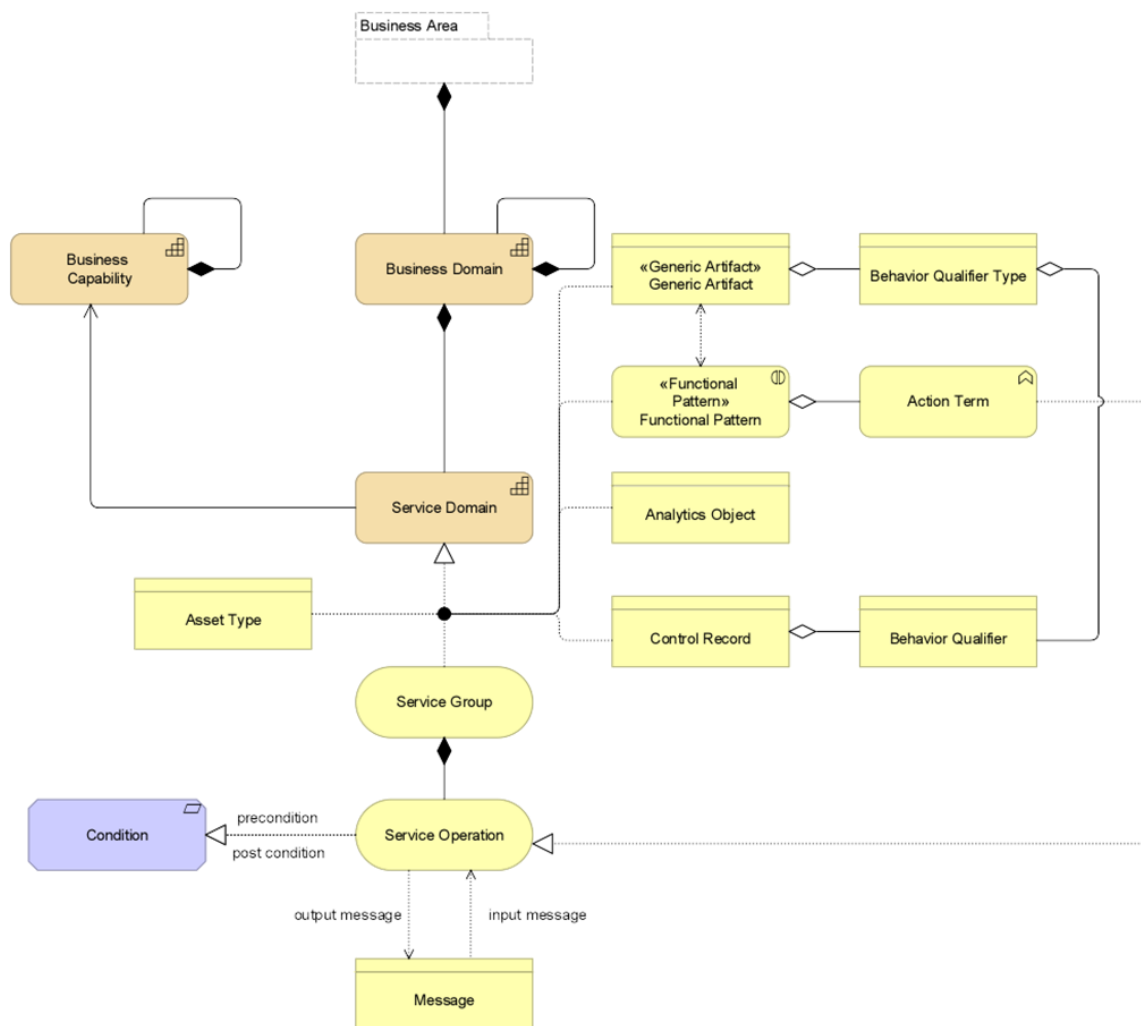


Figure 1: Basic BIAN Metamodel

The BIAN Reference Model is referred to as the “BIAN Service Landscape”.

The BIAN Service Landscape is a reference framework, defining larger business areas and slightly finer-grained business domains into which the full collection of identified service domains can be organized. The key point is that it is intended to be a reference framework for organizing the service domains – it is not intended to represent a design blueprint for any particular type of bank. Such blueprints and the techniques for assembling them are covered elsewhere.

A diagram of the BIAN Service Landscape is shown in Figure 2.²

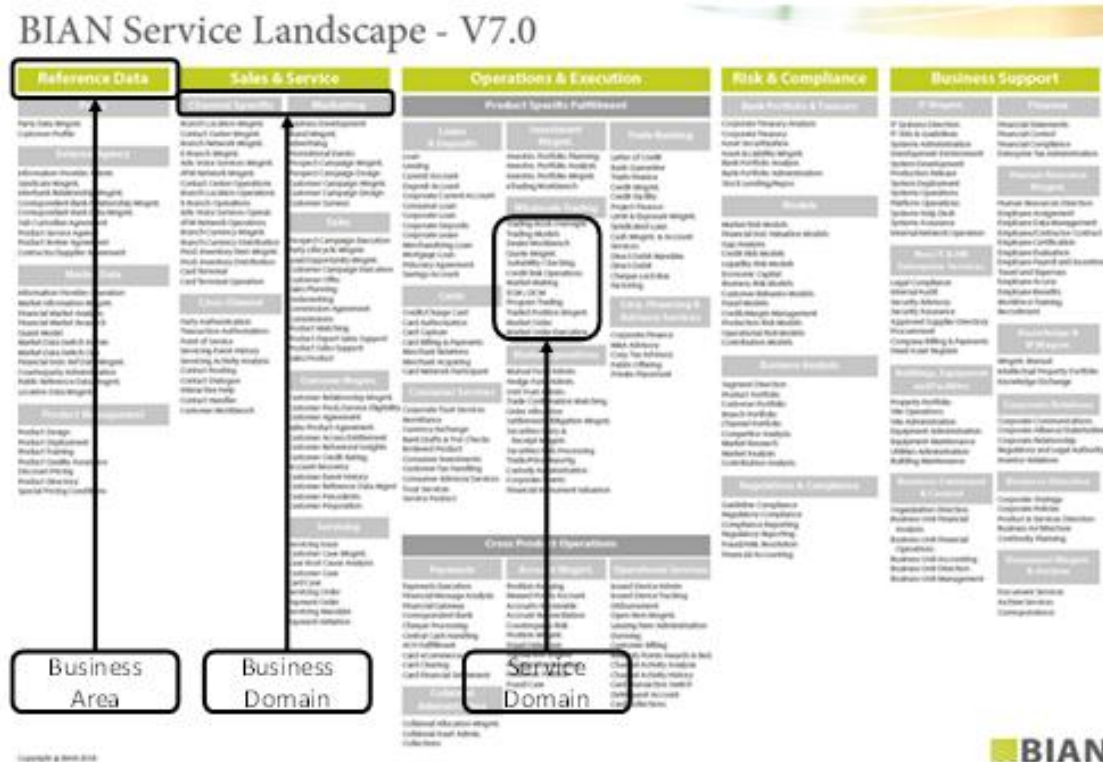


Figure 2: BIAN Service Landscape, Version 7.0

3.1 Business Areas

A business area is formed by a broad set of capabilities and responsibilities and is positioned at the highest level of the BIAN Service Landscape hierarchy. BIAN business areas are used to decompose the functions of financial institutions. This decomposition is primarily driven by business understanding and complemented by business, application, and information-specific needs.

BIAN has identified (amongst others) the following business areas as part of the BIAN Reference Model of business areas and business domains:

- Reference data
- Sales and service
- Operations and execution
- Analytics
- Business support

² Base graphic taken from http://bian.org/wp-content/uploads/2018/11/BIAN_Service_LandscapeV7_0.pdf.

A business area is a collection or a grouping of business domains. It is too coarse-grained to treat it as a capability. It is a concept for navigating through the BIAN Reference Model. In the ArchiMate modeling notation this corresponds to the “grouping” concept.

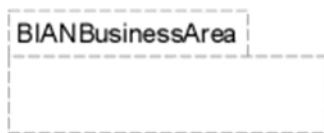


Figure 3: ArchiMate Notation for Business Area

3.2 Business Domains

A business domain is a coarse-grained, non-atomic business capability. It is defined by banking Enterprise Architects to decompose the banking business into a set of mutually-exclusive, collectively exhaustive business capabilities representing the capacity to execute business functions. Business domains are linked to certain skills and knowledge, which are clearly identifiable in the banking business.



Figure 4: Example Business Domains

Figure 4 shows the business area “Operations & Execution” subdivided into a number of business domains. A business domain can be subdivided into more specific business domains. The example only shows a few business domains within the business area. It shows that the business area “Operations & Execution” is subdivided into the business domain “Product-specific Fulfillment”, which in turn is subdivided into three more business domains; “Loans & Deposits”, “Investment Management”, and “Trade Banking”.

A business domain is a coarse-grained functional capability that is of practical use for developing the strategic banking architecture. The business domains define the segments of a bank as seen by banking business architects from a purely functional and architecturally technical perspective.

A business domain corresponds to the ArchiMate capability, as shown in Figure 5.



Figure 5: ArchiMate Notation for a Business Domain

Business domains are part of the BIAN Service Landscape, which is originally represented as illustrated in Figure 2.

- A business domain belongs to exactly one BIAN business area
- A business domain can be decomposed into two or more detailed, non-atomic business domains, as illustrated in Figure 4

The ArchiMate representation of the business domain is illustrated in Figure 6.

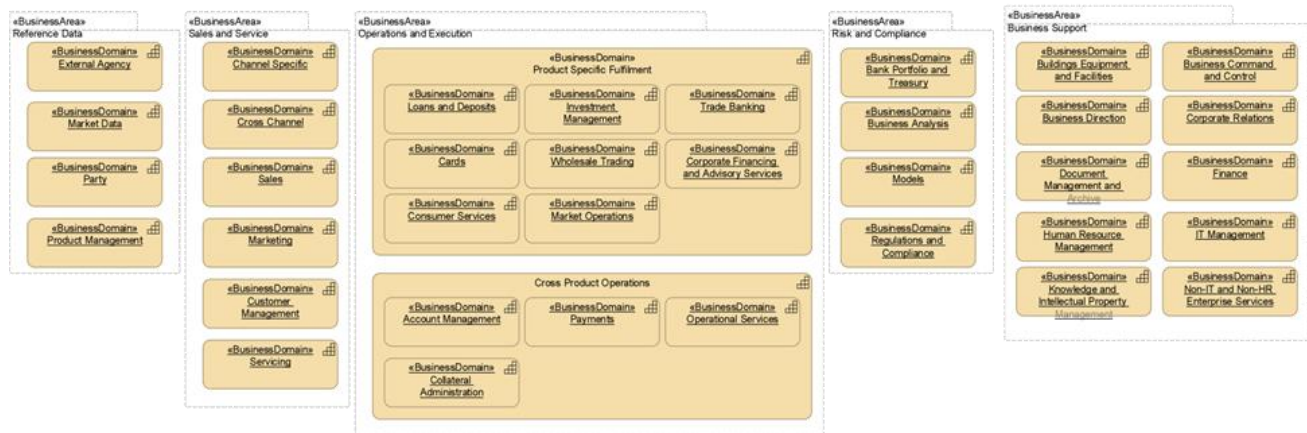


Figure 6: BIAN Service Landscape Overview, Version 7.0 – Business Domains

These coarse-grained “business capabilities” are composed of “atomic” or “elemental” service domains.

3.3 Service Domains

A service domain is an elemental or atomic functional building block within the BIAN Reference Model. A service domain is a capability or capacity to manage the full lifecycle of an asset. It can be seen as a service center with the role to manage the full lifecycle of assets of a certain type.

A service domain is elemental or atomic in scope. This means that the capability is not further decomposed into more detailed capabilities.

Service domains are mutually-exclusive and collectively exhaustive. There is no functional redundancy between the service domains.

A service domain is the most fine-grained capability within the BIAN Service Landscape. A service domain offers its services to other service domains. This allows service domains to fulfill their services by delegating the execution of some functionality to other service domains. The interaction between the service domains realizes the business activities that make a bank a bank.

A service domain corresponds to the ArchiMate capability, as shown in Figure 7.



Figure 7: ArchiMate Notation for a Service Domain

A service domain is a component of exactly one business domain. Figure 8 illustrates the composition of the business domain “Loans and Deposits”.

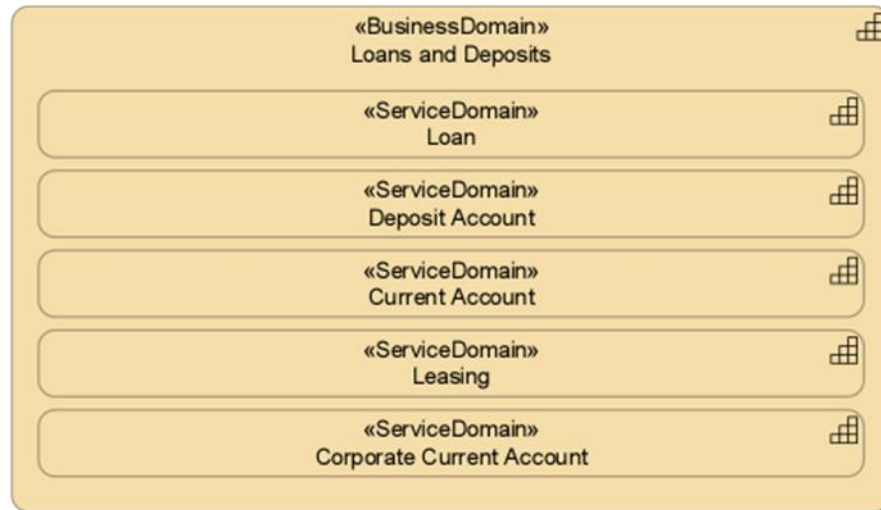


Figure 8: Service Domains of a Business Domain “Loans & Deposits”

Each service domain offers a set of services. These services are called service operations. A service domain is a set of service operations that together manage the full lifecycle of an asset.

A BIAN service domain combines an asset and a use. The technique used to isolate a BIAN service domain is to apply a functional pattern on an asset-type. A functional pattern is a set of predefined and standardized action terms that will operate on the asset type.

BIAN has defined functional patterns, action terms, and a hierarchical classification of the assets (tangible and intangible) that may make up any bank.

Each service domain combines a single primary functional pattern (for example, maintain reference details, or define and execute a plan) with an asset or entity type (for example, a piece of equipment, or a customer relationship).

These building blocks of the service domain are represented in Figure 9 and will be explained in more detail in the following sections.

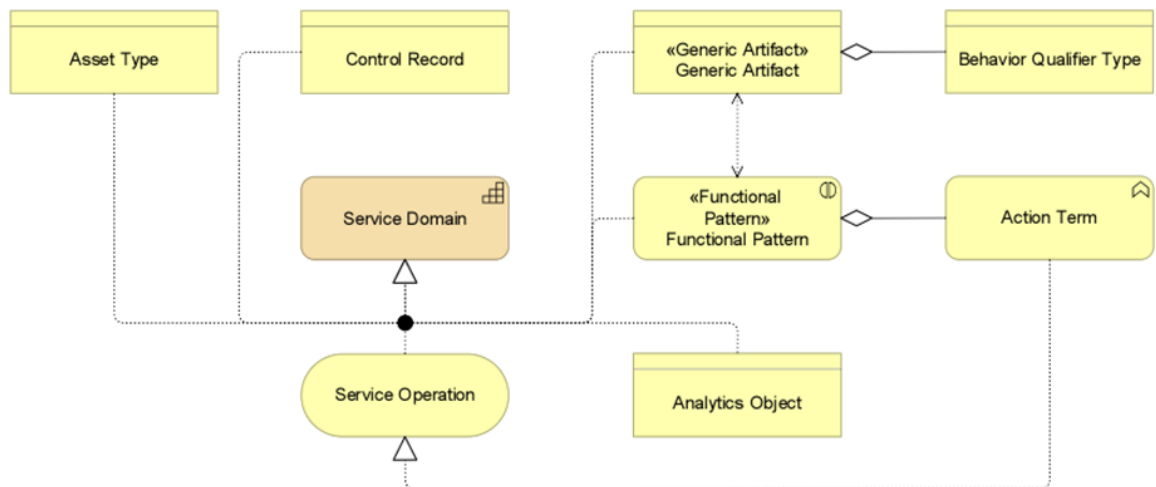


Figure 9: BIAN Metamodel Focusing on a Service Domain

3.3.1 Action Terms

BIAN has identified a standard set of actions that characterize the different types of service operations. Each service operation executes exactly one action. There are currently 16 standard detailed actions terms.

The full list of action terms (represented as ArchiMate business functions) is shown in Figure 10.

An action term is represented by an ArchiMate business function.



Figure 10: Action Terms

3.3.2 Functional Pattern

A collection of action terms that together form a recurring type of behavior is called a functional pattern. BIAN, Version 7.0 identifies 18 functional patterns, as shown in Figure 11.

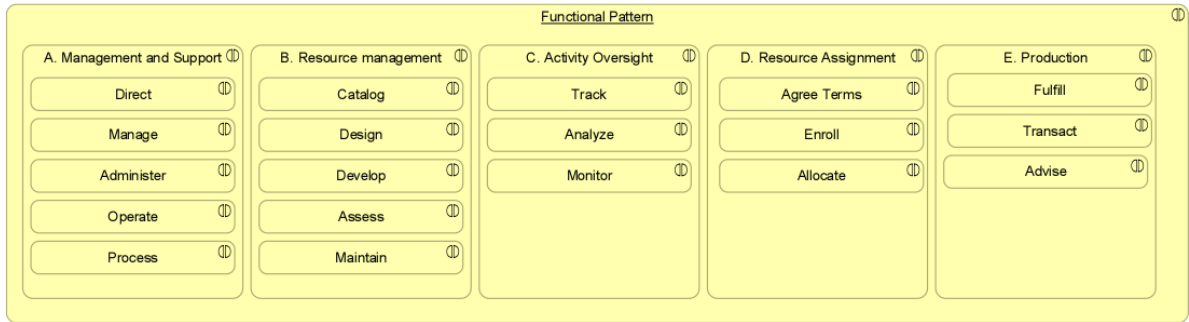


Figure 11: Functional Patterns

The set of action terms that make up the functional pattern are visualized in the functional pattern matrix, as shown in Figure 12.

A functional pattern corresponds to the ArchiMate business interaction.

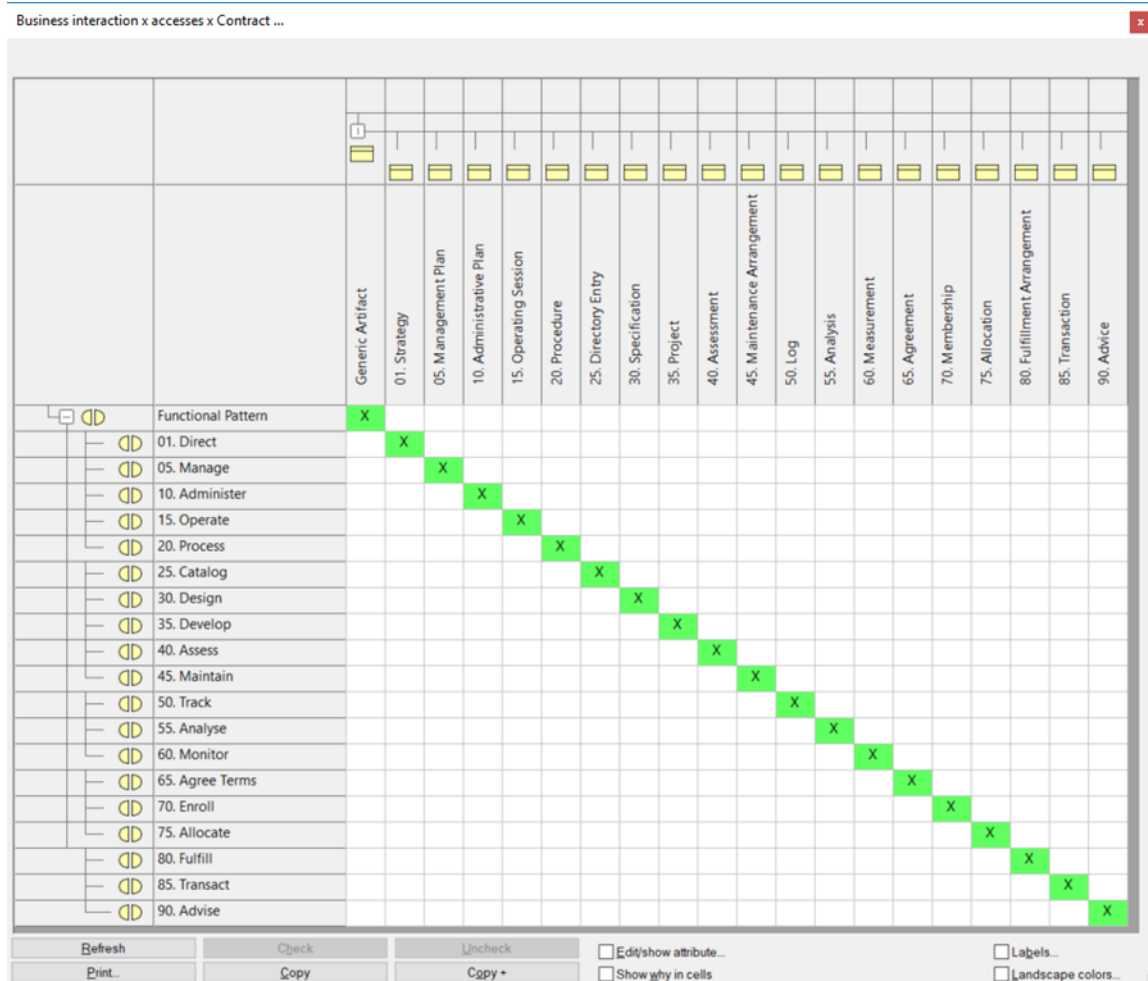


Figure 13: Functional Pattern and Generic Artifact

A control record represents the information required for a service domain to deliver its services. It is the log of information about the lifecycle of an asset type when accessed by a functional pattern according to, or resulting in, an instance of a generic artifact.

If, for example, the asset type is “Current Account”, the functional pattern is “Fulfill”, and the coupled generic artifact is “Fulfillment Arrangement”, then the specific control record is “Current Account Fulfillment Arrangement”.

The name of the control record is the concatenation of the name of the asset type and the name of the generic artifact. The service domain “Current Account” will deliver services related to the “Current Account Fulfillment Arrangement”.

A control record can be seen as the information of the lifecycle of a “qualified asset type” where the qualifier is the generic artifact. It is represented using the business object component in the ArchiMate language.

3.4 Service Operations

A service operation is the externalization of an action upon a qualified asset type. It is an elementary service.

In the example of the service domain “Current Account”, the service domain is capable of performing “initiateCurrentAccountFulfillment”, “executeCurrentAccountFulfillment”, etc. which are the action terms aggregated in the functional pattern and applied to the control record.

The “Current Account” fulfillment services, or service operations, are derived from the functional pattern action terms. Services are organized into service groups. The service groups are standardized as:

1. Setup
2. Origination
3. Invocation
4. Reporting activities

In the example of the service domain “Current Account”:

- Setup of the service domain “Current Account”:
 - Record
- Origination of the “Current Account” fulfillment:
 - Initiate
- Invocation of the “Current Account” fulfillment:
 - Execute
 - Request
 - Terminate
 - Update
- Reporting of the “Current Account” fulfillment:
 - Notify
 - Retrieve

Service operations are represented as ArchiMate business services.

An overview of the service domain can be represented as shown in Figure 14.

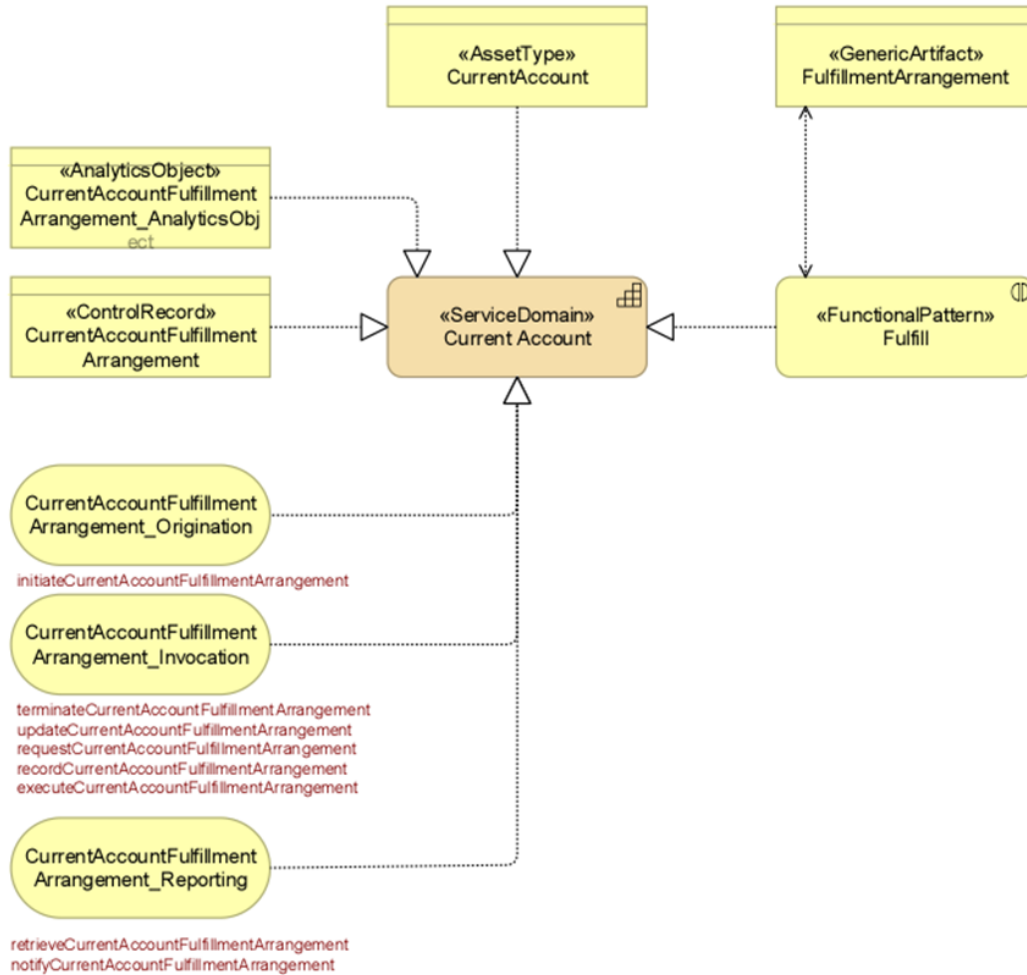


Figure 14: Overview Diagram of the Service Domain “Current Account”

3.4.1 Input/Output Message

Service operations within a service domain can be accessed through clearly-defined interfaces. Every service operation needs a minimum input to be able to deliver the service. This input is a message type which is called the input message. The service will be realized by some internal processing, possibly delegating some tasks to other services. As a result, the service will deliver a result that is expressed in an output message. The concept of the message is visualized in the metamodel diagram shown in Figure 15. When applying this metamodel to create a model, the business object “Message” will be made specific by named messages such as “Invoice” or “Payment Request”. A message which is an input message for one service operation can be the output message from another service operation.

Messages are represented as ArchiMate business objects.

3.4.2 Pre/Post-Condition

Before a service starts the processing of fulfillment, it will evaluate if the conditions are fulfilled in order to accept and start the service. These conditions that are controlling the status at the beginning of the process are called the pre-conditions. This is not the only check that is

performed. The service will also check if the status of the involved objects at the end of the process is acceptable. These conditions are called the post-conditions. These conditions are illustrated in Figure 15.

Pre/post-conditions are represented by ArchiMate requirements.

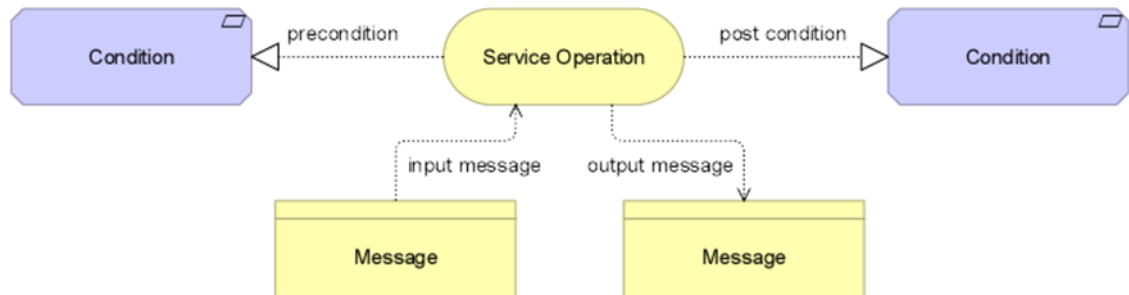


Figure 15: Service Messages and Conditions

4 BIAN Business Scenarios and Wireframes

BIAN offers business scenarios and wireframes, a process-oriented view on the reference architecture.

A service domain delivers multiple business services. Business scenarios are the process-oriented realizations of services of several service domains. Business scenarios are the interaction of service operations. These service operations interact via the service operation connections to realize the desired outcome of the scenario. Service operations serve one or more business scenarios. Figure 16 shows the metamodel for business scenarios. As the ArchiMate Specification makes no distinction between concepts and the instances of the concepts, the service operation visualized in the diagram needs to be interpreted as a set of service operations that serve the business scenarios. Service operation connections are the serve relations between service operations. Before a scenario starts executing, all conditions need to be fulfilled. Conditions at the start of the scenario are called the pre-conditions. Conditions at the end are called the post-conditions.

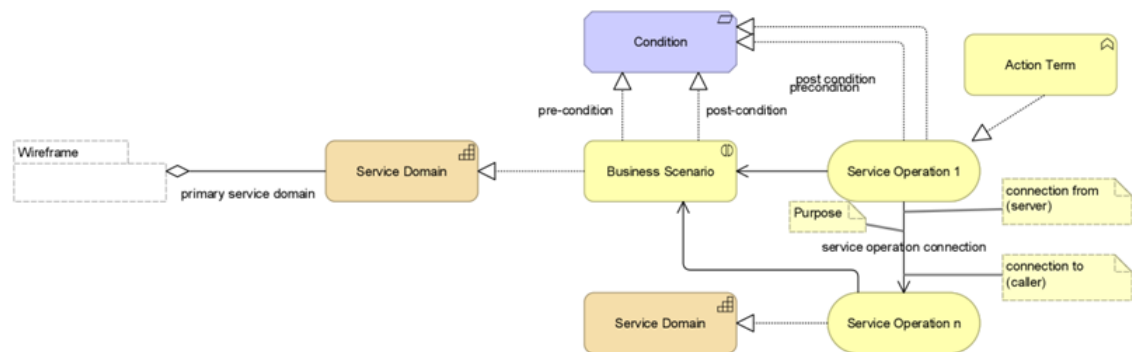


Figure 16: BIAN Model for a Business Scenario

4.1 Business Scenarios

The dynamics of banking are in the offering and delivery of banking services. In a service-oriented banking environment the banking services are realized by an interaction of multiple service operations, together delivering the required outcome.

This interaction of service operations is called a business scenario.

The scope of a business scenario can be compared to that of a conventional high-level business process with one key difference: both describe action steps and some implicit flow of control, but, unlike the business scenario, the business process does not formally divide functionality between discrete service-based partitions (service domains).

In BIAN there is not really a focus on the sequence in which these service operations execute, although BIAN published sequence diagrams to give an idea of how a service can be fulfilled. It explains the context of the interactions. In a service-oriented banking environment the focus is

on the interaction, the interoperability of services to exchange information, and information services between the service operations of the service domains. This is expressed in so-called wireframes.

A business scenario is visualized in a UML sequence diagram. An example is shown in Figure 17.

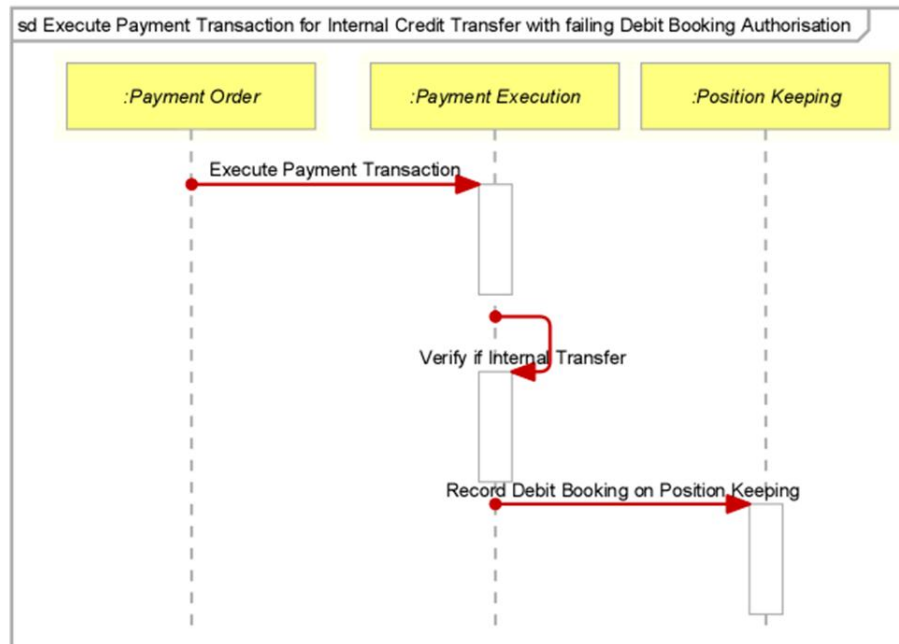


Figure 17: Sequence Diagram Internal Credit Transfer

At the time of writing, BIAN will not translate this into ArchiMate notation; i.e., a process diagram with service calls.

4.2 Wireframes

A wireframe model is a static model, showing the service domains and available connections. Conversely, a business scenario is a dynamic model that shows the temporal pattern of a collection of interactions that are triggered by some business action or event.

Figure 18 is an example of a wireframe showing the main service operations for a collection of service domains involved in “Consumer Loan – Interest & Redemption, Collections & Customer Risk Revision”.

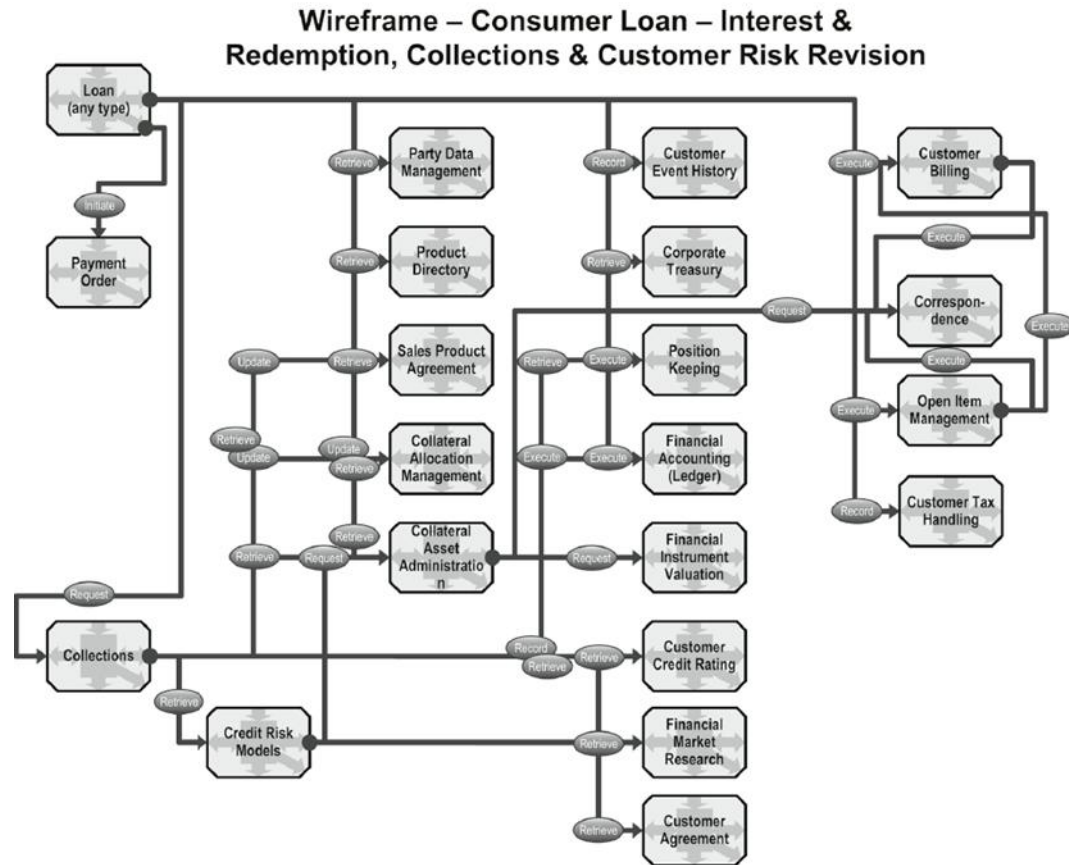


Figure 18: Wireframe for a Collection of Service Domains Involved in Consumer Loan

At the time of writing, BIAN has not yet translated wireframes into ArchiMate notation. An example of how a wireframe might look in ArchiMate notation is given in Figure 19.

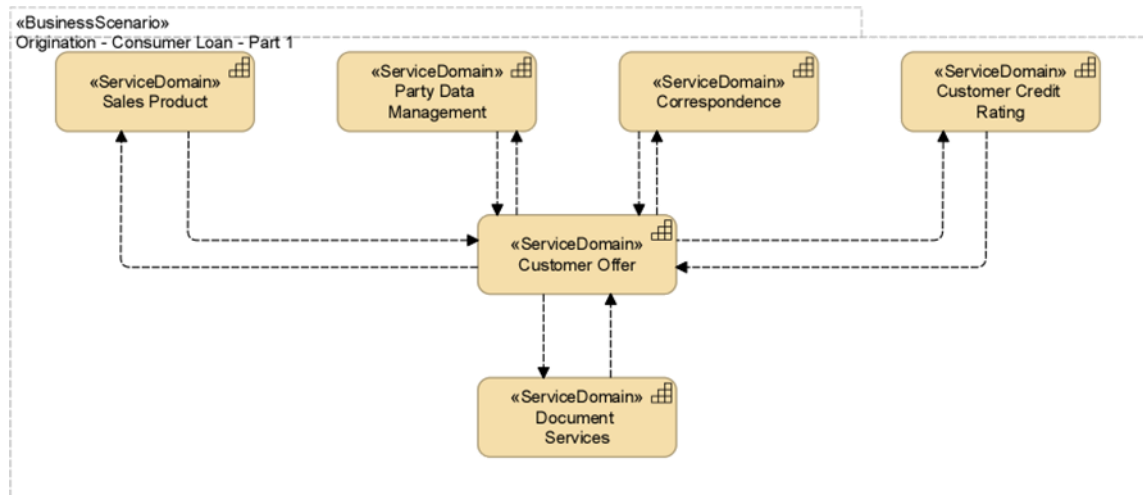


Figure 19: Wireframe Consumer Loan (Partial) in ArchiMate Notation

These components can now be used to define an organization-specific banking Enterprise Architecture landscape.

5 BIAN Business Object Model Framework

5.1 BIAN Business Object Model

In today's competitive marketplace, executive leaders are racing to convert data-driven insights into meaningful results (IBM Institute for Business Value; see [Referenced Documents](#)). A Data Architecture Management capability³ is now the key differentiator in creating value for businesses. A qualitative Enterprise Information Architecture (EIA) is a prerequisite for qualitative Data Architecture Management.

Recently, BIAN is applying the Business Object Modeling (BOMing) approach in the banking domain to develop a Business Object Model (BOM) for the financial industry (BIAN BOM).

The existing banking data models and standards such as ISO 20022, IFX, and SWIFT™ are focused on defining the messages exchanged between Application Program Interfaces (APIs). These messages are views on the core business objects in banking (such as bank products and services, bank agreements and arrangements). The existing standards do not focus on structuring and defining these information building blocks in any structural or understandable way for business people. BOMing creates a precise conceptual basis for BIAN to fill this gap in understanding business semantics (business concepts). It supports the creation of a BOM as part of the Information Architecture. The BIAN BOM provides the financial sector with a reference model for Information Architecture that can be customized to individual needs.

The BOMing approach is being applied to model the information needs of every BIAN service domain. The resulting service domain models are integrated into the BIAN BOM, a reference model for the EIA of the financial industry.

The ownership⁴ of each business object is attributed to one service domain.

The BIAN business objects will be detailed into fully attributed information entities, accompanied by unambiguous definitions. As such, the control records of the BIAN service domains, containing the information required for the service domain, and the input/output messages of the service operations, will be defined as views on the BIAN BOM, as represented in Figure 20.

³ Data Architecture Management provides the ability to address, for example, data standardization, centralization, extracting, transformation, movement, storage, integration, and governance.

⁴ The service domain responsible for the quality of the information and for the provision of information services.

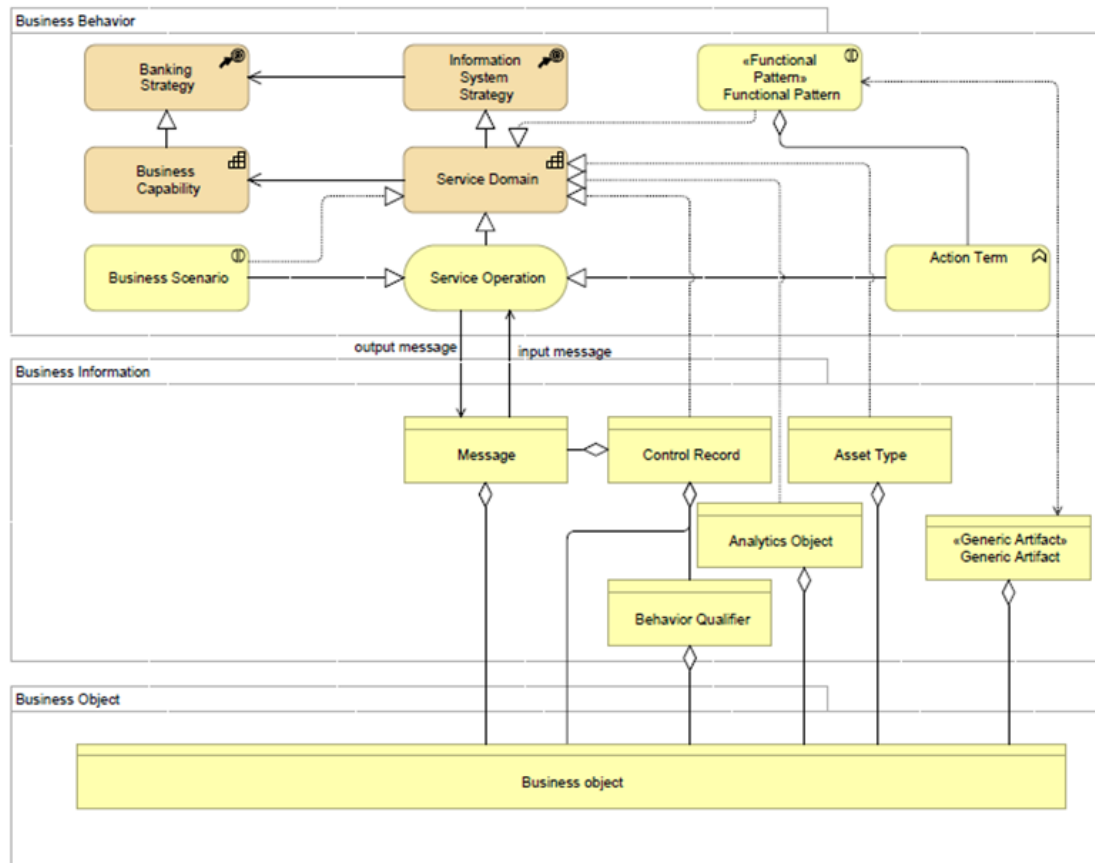


Figure 20: Business Objects in the BIAN Metamodel

The BIAN BOM business objects are expressed as ArchiMate business objects.

5.2 Business Object Modeling: The Approach

BOMing is a way of thinking, complemented by two abstract reference models that are used as patterns for the actual BOM of an enterprise or domain.

In the following sections, we will explain the way of thinking and the content and structure patterns that constitute the BOMing approach.

5.2.1 Business Object *versus* Business Concept

Two key notions in the BOMing way of thinking are “business concept” and “business object”. Being able to make the distinction is key.

A concept is whatever can be thought of. A business concept is a concept that is of importance to the business; i.e., something business wants to be informed about.

In order to fulfill the information requirements of the business, business concepts need to be identified. Each business concept should be defined unambiguously by means of a business definition, in order to avoid misunderstandings between involved parties. Each business concept

will be referred to by one or more names.⁵ The name of a business concept is often called the business term. The business term is a representation of a business concept by means of a word or ordered set of words.

The business glossary contains the business concepts, ordered by business term and definition.

In order to inform businesses concerning the concepts in which they are interested, data needs to be captured and managed. Business concepts, however, are not the building blocks for the Information Architecture required to steer an effective Data Architecture.

The building block of the business Information Architecture is the business object. It is a mutually-exclusive, collectively exhaustive unit of information. Business objects relate to each other, and thus constitute the BOM.

A business object is also business concept, or an abstraction thereof;⁶ hence, it also needs to be named by a business term and unambiguously defined in business words. It is also part of the business glossary.

The BOMing approach presented here is used to produce unambiguous definitions for business concepts; a prerequisite for distinguishing business objects from business concepts that are “views”.

For example, “customer” is clearly a business concept. Any organization will be interested in the parties it serves. However, it is not a business object. A “customer” is a “party” (business object) that buys “products”. The same party can have a “supplier” and “co-worker” role.

5.2.2 Content Pattern

The BOM content pattern, represented in Figure 21, is an abstract information model, valid for any business.

It is made more specific for a specific business context or domain.

Only the core business objects that are most important for a financial institution are described here:

- Businesses distinguish themselves from others by the products (type of goods and services or a coherent package thereof) they offer to internal or external customers; a service is a valuable functionality that fulfils a need or a requirement
- The sale of a product or provision of a service is concluded by an agreement with the customer – agreements with suppliers and authorities enable the business to offer its products to the market

An agreement is a formal or informal common understanding between two or more parties, concerning one or more subject matters expressed in a set of arrangements, terms, and conditions.

⁵ One name will be selected as the “business term”, the others can be managed as synonyms.

⁶ A business object is a business concept, or becomes one; e.g., “customer”, “correspondent”, and “supplier” are commonly business concepts. Abstraction results in the “party” and “party role” business objects (that can have the value “customer”, “correspondent”, and “supplier”). These will soon enough become business concepts.

- An agreement is composed of a number of arrangements, where one party engages him/herself against another party to give, to do, or not to do something (the subject of the arrangement)

An arrangement always consists of:

- A “subject matter” which can be every “thing”
- An “act” to do, not to do, to give or not to give some “thing” (the subject matter)
- Parties: it is a “party” who makes the promises to do, not to do, to give or not to give something for/to a “counter party”

- A party is an individual or an organization

Parties are of interest to a business because of their (potential) involvement in agreements.

- A (party) role is the responsibility or involvement of a party in a specific business context

Together, they form a balance to which these parties commit to achieve. Agreements and arrangements are the core notions of a “commitment system”. This system manages the promises that a business makes toward its stakeholders and ensures promises can be fulfilled. A commitment system does not in itself fulfill any promises or commitments. It makes certain that it is possible to do so. “Commitments” or “promises” are the source of income for a business according to Knaepen and Brooms, 2013 (see [Referenced Documents](#)).

- The fulfillment of one or more arrangements of an agreement can be triggered by giving an instruction to do or to give something – an instruction is a request to do something; the commitment made in an arrangement

The instruction will be accepted for execution only if the conditions, agreed upon in the agreement, are met.

- Instructions trigger transactions needed to fulfill the arrangements

Instruction and transaction are the core notions of a “Fulfillment Management System”, where the commitments made are actually fulfilled.

A transaction is the act to do “something”. It can either be the conclusion of an agreement (establishing arrangements), or the execution of the commitments specified in an arrangement.

- When a transaction affects the position on an account, an account entry must be made on the appropriate account

- An account is a measuring state on which movements in value or amounts of assets, rights, and obligations are registered

A financial account is a specialization of “account”: an administrative financial state where the amounts of financial transactions are registered in debit or credit, resulting in a balance.

- An account entry is the record of a movement in value or amounts of assets, rights, and obligations

A financial account entry (or booking) is the record of a financial transaction on a financial account. It results in an increase or decrease of a balance.

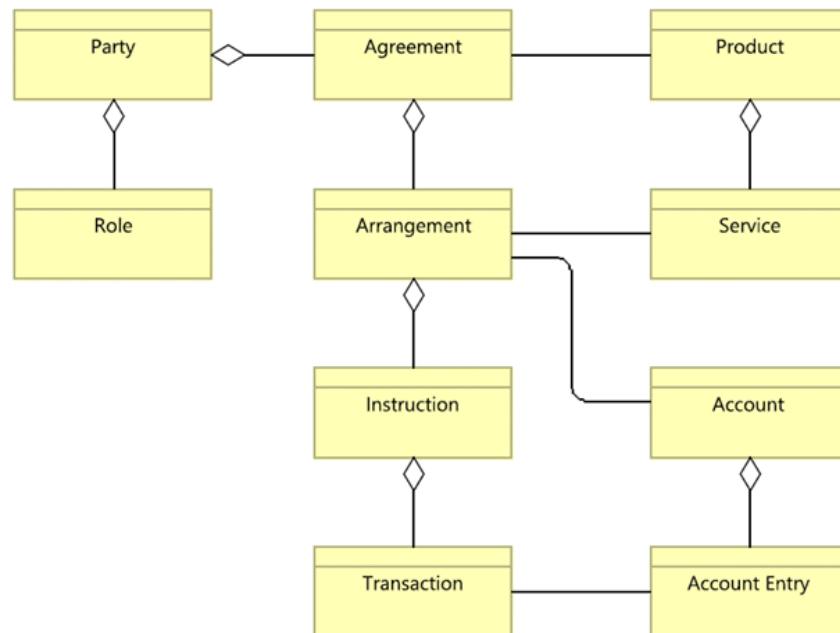


Figure 21: BOM Content Pattern

5.2.3 Structure Pattern

The BOM structure pattern (represented in Figure 22) enriches the content pattern.

A business object can be subject to classifications, resulting in a “business object type” business object, that contains the classification classes (e.g., a party can be of “gender type” male, female, or X).

A business object will most probably have relationships with other business objects, resulting in a “business object relationship” object (e.g., a party will have involvements in agreements). This relationship object can in turn be classified by a “business object relationship type” object (e.g., involvement types “buyer”, “seller”, “sales person”, etc.).

A business object is described by “business object descriptors”, of a “business object descriptor type” – which will not appear in the BOM diagram, but rather as attributes of the information entities, detailing the BOM.

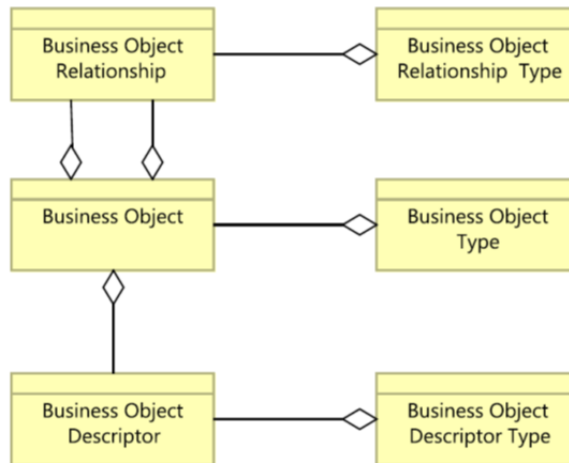


Figure 22: BOM Structure Pattern

The BOM content pattern, enriched by the BOM structure pattern, provides a powerful support for BOMing. Firstly, because it provides a powerful support for an unambiguous definition of a business concept, and secondly, because it provides a completeness check for each model.

Thus, both the quality and speed of the Information Architecture work will be improved.

A modeler systematically asks the following questions:

- What is the business concept?

Is it a (specialization of) a content pattern object (a party, agreement, or arrangement)? Or is it a classification thereof? Or maybe a relationship between objects or classifications thereof? Or a descriptor?

Thus, the structure pattern supports the derivation of the BOM from the multitude of business concepts relevant for the business domain.

- What (specializations of) content pattern business objects are relevant for the business context?

How are the content pattern relationships specialized here? What other relationships exist between the relevant business objects? What classifications are relevant?

Thus, the content pattern, extended by the structure pattern, supports the completeness of the BOM.

- What are the descriptors (e.g., identifiers, lifecycle status, other attributes)?

Do we have descriptors of all relevant types?

Thus, the structure pattern supports the completeness of the detailed information model.

5.2.4 Classification Mechanism

Classification is a mechanism to order or group business concepts according to specified criteria. Business concepts can be classified according to multiple viewpoints, because different stakeholders look at business concepts differently. These viewpoints may depend on function or profession (for example, finance has a different view on a customer than marketing) or level in

the organization (the board of directors will classify company activities differently from a shop worker).

From an Information Architecture perspective, it is important to distinguish two types of classifications:

- A taxonomical classification is a mechanism to classify concepts from the perspective of the nature of the concept

A concept can only be classified to one and only one class of the taxonomical classification. The characteristics that are used in the rules categorizing the instances of the concepts into a taxonomical class are inherently describing the business object. For example, individuals can be classified according to their gender as a “male” or “female”. Each instance can only be classified as “male” or “female”. Taxonomical classifications will require a “type” object, but can also lead to specializing business objects, as they are based on specific characteristics.

- A functional classification is a mechanism to classify concepts from the perspective of functional interest

Most classifications are from a functional perspective. For example, an individual can be classified according to their role in an agreement; e.g., buyer or seller. A “buyer”, as a party, does not have specific characteristics distinguishing them from a “seller”. Hence, a functional classification will result in a “type” object, but not in a specialization.

5.2.5 Example Payments

In a credit transfer, several service domains are involved. We will simplify matters by looking solely at the current account of the payment originator (i.e., the party that has the role “payment originator” in the payment order).

The service domain “Current Account” is the owner of the “Current Account Agreement” information,⁷ which bundles arrangements regarding, for example, the promise of the bank to transfer funds when asked to do so (i.e., the “Credit Transfer Service Arrangement”) and the related promise of the customer to pay for that transfer according to the “Pricing Arrangement”. This service domain makes credit transfers possible (or impossible, if the “Current Account Agreement” is blocked).

The “Payment Order” service domain checks the acceptability of an instruction to transfer funds; i.e., it checks the conditions for the fulfillment of the promises made in the agreement.

The “Payment Execution” service domain takes care of the transactions required to execute the “Payment Order”.

One of the account entries required to adequately register these transactions is the “Current Account Booking” on the “Current Account”.⁸

⁷ The “Current Account Agreement” is the result of a sale (concluded by the “Sales” service domain of the “Current Account” product, that was engineered by the “Product Design” service domain.

⁸ The “administrative financial state where the amounts of financial transactions are registered in debit or credit, resulting in a balance”.

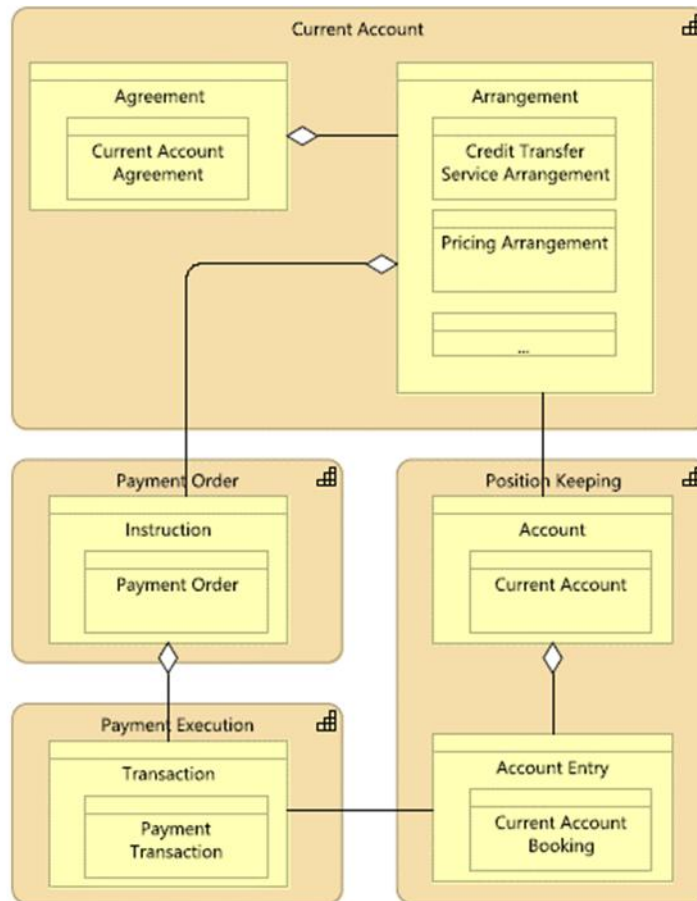


Figure 23: Applying the BIAN BOM in Payments

6 Case Study: Archi Banking Group

The Archi Banking Group Case Study is a fictitious example developed to illustrate the use of the BIAN Reference Model, ArchiMate modeling notation, and the TOGAF® framework.

The BIAN Reference Model is meant to be used as a reference to develop a bank-specific Enterprise Architecture.

An example use of the BIAN Reference Model, expressed in the ArchiMate language while applying the TOGAF framework, is described in the Case Study: Archi Banking Group (see [Referenced Documents](#)). This is a fictitious example inspired by a real-life use-case; i.e., the governance of a corporate program portfolio for payment change initiatives.

The Case Study describes a banking group composed of banks and legal entities in different countries. Each of the legal entity banks are working according to a group strategy, although each of the banks has their own identity and, as such, their own strategy. The dynamics of portfolio management are described using the federated architecture approach. The Case Study covers all the architecture layers from strategic goals and strategy, the architecture landscape definition, opportunities and solution, migration planning, and implementation governance, and the need to adapt the banking architecture via architecture change management, in a segmented architecture Enterprise Architecture organization. It demonstrates how the ArchiMate Specification, the TOGAF framework, and BIAN can be used to create high-quality architectures. It can also be used as an inspiration to apply the combination of the three standards in real-life banking.

Index

account	30	EIA	27
account entry	30	financial account entry	31
action term.....	18	Fintechs	10
agreement	29	functional classification	33
ArchiMate language	9	functional pattern	18
arrangement.....	30	functional pattern matrix	19
asset type	20	generic artifact.....	20
best practices	10	input message.....	22
BIAN	9, 11	instruction.....	30
BIAN BOM	27	ISO 20022	9
BIAN Reference Model.....	11	output message	22
BIAN Service Landscape ...	11, 13	party	30
BOM.....	27	post-conditions	23
BOMing.....	27, 28	pre-conditions.....	23
business area.....	14	product	29
business concept.....	28	role	30
business domain	15	service	29
business glossary	29	service domain	16
business object.....	29	service groups	21
Business Object Model.....	27	service operation	17, 21
Business Object Modeling.....	27	specialization.....	11
business scenario	24	standard interfaces.....	10
business term	29	structure pattern.....	31
classification.....	32	taxonomical classification.....	33
commitment system.....	30	transaction	30
content pattern	29	UML.....	9, 25
control record	20	wireframe	25
Data Architecture Management	27		