HOPEX Archimate

User Guide

HOPEX V2

INTRODUCTION TO HOPEX ARCHIMATE

HOPEX Archimate is a full-web implementation of the Open Group's ArchiMate 2.1 Enterprise Architecture standard http://www.opengroup.org/archimate/, which provides an implementation of a growing interest Enterprise Architecture standard published by the Open Group, who also edits TOGAF, covering all the layers of Enterprise Architecture.

HOPEX Archimate provides a metamodel and a notation covering the Business, Application and Technology Enterprise Architecture layers according to the ArchiMate framework, whilst remaining compatible with other **HOPEX EA** products.

HOPEX Archimate implementation focuses on the core ArchiMate viewpoints; the ArchiMate 2.1 extensions being already covered as a standard by **HOPEX Business Process Analysis** (for the motivation extension) and **HOPEX Planning** (for the implementation and deployment extension).

FEATURES SUMMARY

HOPEX Archimate provides:

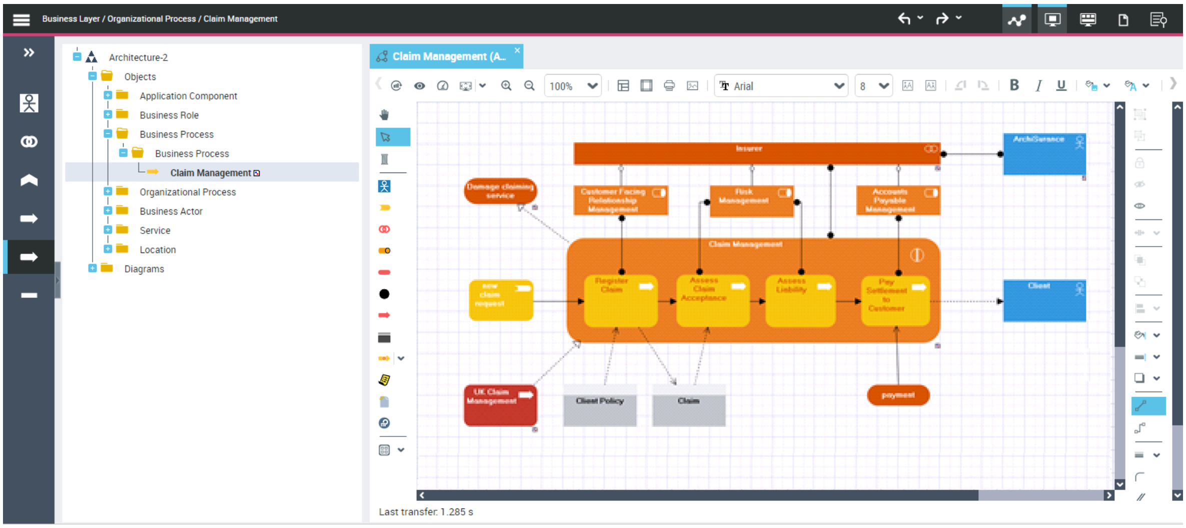
- The main core concepts and viewpoints of the ArchiMate 2.0 Open Group Standard;
- A dedicated "Viewpoints oriented" full-web desktop granting an easy access to the EA data for the end user;
- New flat design based shapes and icons;
- The usage of the ArchiMate vocabulary and pictures for the ArchiMate user profiles, whilst using the same underlying **HOPEX** concepts as in the other **HOPEX** products, so that each user can share the same EA assets with other users while keeping its own vocabulary and pictures;
- Compatibility with other **HOPEX** products, enabling a model to be drawn in a given notation (e.g. a process in ArchiMate) and detailed in another (e.g. BPMN)

Dedicated Viewpoint Desktop

HOPEX Archimate provides a "viewpoint" oriented desktop which has been specifically fine-tuned for ArchiMate. This desktop is dynamically build based on the definition of Viewpoints as part of an "ArchiMate" Framework.

The desktop entry point is a new "Architecture" concept which gathers objects of interest for a user or a set of users; the Architecture is created according to a Framework, and the Architecture elements can be browsed according to defined Viewpoints.

Each viewpoint is associated with diagram types, report types and metaclasses which are of interest for the viewpoint, enabling the user to easily access a given diagram and to easily enable drag and drop of relevant objects of an architecture on this diagram.



***HOPEX Archimate** viewpoint web desktop*

About This Guide

This guide presents how to make best use of **HOPEX Archimate** to assure efficient management of your risks.

Guide Structure

The **HOPEX Archimate** guide comprises the following chapters:

- "The HOPEX Implementation of ArchiMate", page 11: describes the main principles that govern the ArchiMate implementation of **HOPEX Archimate**.
- "Discovering the Desktop", page 35: presents the interface.
- "HOPEX Archimate ViewPoints", page 45: presents the functions offered by **HOPEX Archimate** to model the aspects of your enterprise architecture.
- "The HOPEX Metamodel for ArchiMate", page 83: describes the **HOPEX** metamodel used to implement **HOPEX Archimate**.

Additional Resources

This guide is supplemented by:

- The **HOPEX Common Features** guide, which describes basic functions common to **HOPEX** solutions.

  ☛ *It can be useful to consult this guide for a general presentation of the interface.*
- The **HOPEX Assessment** guide, which describes functions proposed by **HOPEX** to use and customize assessment questionnaires.
- The administration guide **HOPEX Power Supervisor**.
- more advanced technical functions are described in the **HOPEX Power Studio** guide.

Conventions Used in the Guide

Styles and formatting

☛	*Remark on the preceding points.*
📖	*Definition of terms used.*
☺	*A tip that may simplify things.*
🦕	*Compatibility with previous versions.*
💣	**Things you must not do.**

💣 **Very important remark to avoid errors during an operation.**

Commands are presented as seen here: **File > Open**.

Names of products and technical modules are presented in bold as seen here: **HOPEX**.

The HOPEX Implementation of ArchiMate

HOPEX proposes an implementation based on the ArchiMate 2.1 specification which aims at ensuring continuity with other **HOPEX** products such as :

- ✓ **HOPEX Business Process Analysis** for the business layer,
- ✓ **HOPEX Logical Data** and **HOPEX Information Architecture** for the information and data sub layers,
- ✓ **HOPEX System Oriented IT Architecture**, for the application layer,
- ✓ **HOPEX IT Architecture** for the technology layer.

☛ *It is a partial implementation in terms of concepts or links proposed by ArchiMate, and the extensions are not covered as part of this release. The key objective was to propose the core of the ArchiMate notation whilst being consistent with the overall **HOPEX** EA stack.*

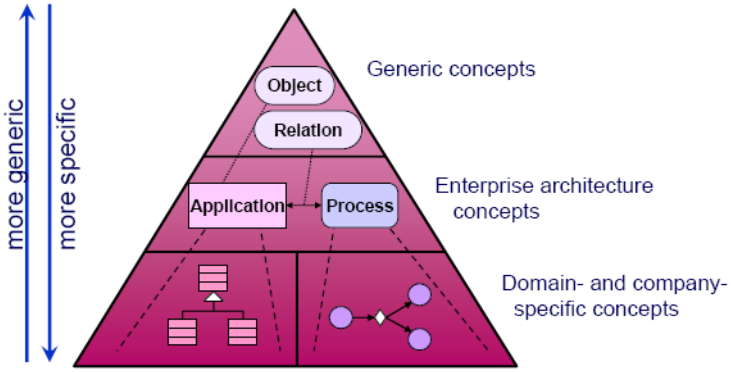
The aim is to present the main principles that govern this implementation and guide the user in his/her use of the **HOPEX** Modeling tool to create ArchiMate deliverables.

ArchiMate language structure

This chapter presents the construction of the ArchiMate architecture modeling language. It discuss some general ideas, principles, and assumptions underlying the development of the ArchiMate metamodel.

Design Approach

A key challenge in the development of a general metamodel for enterprise architecture is to strike a balance between the specificity of languages for individual architecture domains, and a very general set of architecture concepts, which reflects a view of systems as a mere set of interrelated entities. Figure below illustrates that concepts can be described at different levels of specialization.



ArchiMate Metamodels at different levels of Specify

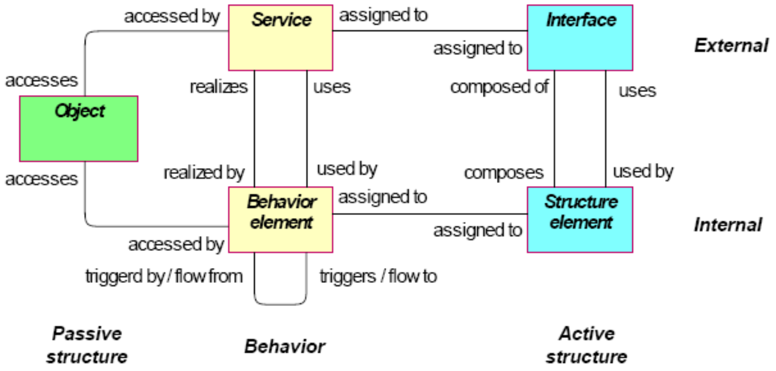
At the base of the triangle we find the metamodels of the architecture modeling concepts used by specific organizations, as well as a variety of existing modeling languages and standards; UML is an example of a language in this category.

At the top of the triangle we find the "most general" metamodel for system architectures, essentially a metamodel that merely comprises notions such as "object", "component", and "relation". The design of the ArchiMate language started from a set of relatively generic concepts (higher up in the pyramid). These were then specialized towards application at different architectural layers, as will be explained below.

Core Concepts

The language consists of active structure elements, behavioral elements and passive structure elements.

- The active structure elements are the business actors, application components and devices that display actual behavior, i.e., the 'subjects' of activity (right side of the Figure).
- Then there is the behavioral or dynamic aspect (center of Figure). The active structure concepts are assigned to behavioral concepts, to show who or what performs the behavior.



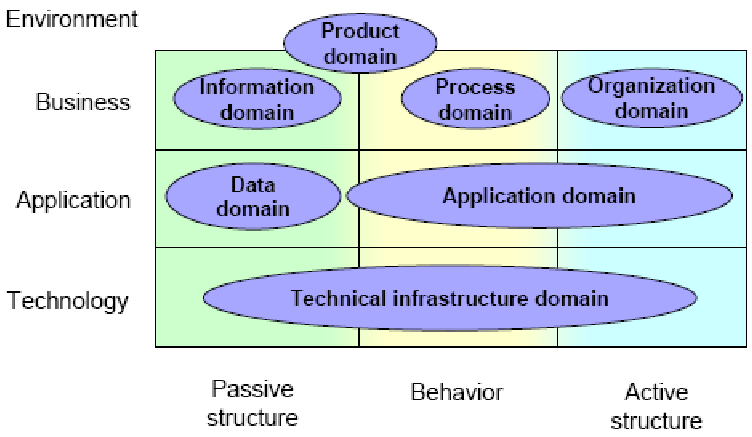
ArchiMate Generic Metamodel : The core concepts

- The passive structure elements are the objects on which behavior is performed. In the domain of information-intensive organizations, which is the main focus of the language, these are usually information or data objects, but they may also be used to represent physical objects.

These three aspects – active structure, behavior, and passive structure – have been inspired by natural language, where a sentence has a subject (active structure), a verb (behavior), and an object (passive structure).

The ArchiMate Framework

It is important to realize that the classification of concepts based on conceptual domains, or based on aspects and layers, is only a global one. It is impossible to define a strict boundary between the aspects and layers, because concepts that link the different aspects and layers play a central role in a coherent architectural description. For example, running somewhat ahead of the later conceptual discussions, (business) functions and (business) roles serve as intermediary concepts between “purely behavioral” concepts and “purely structural” concepts.



ArchiMate Architectural Framework

ARCHIMATE LAYERS AND RELATIONSHIPS

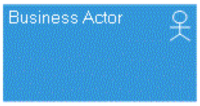
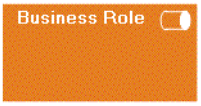
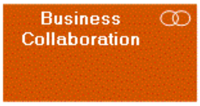
This chapter precise definition and illustration of its generic set of concepts of ArchMate 2.1. They provide a proper basis for visualization, analysis, tooling, and use of these concepts.

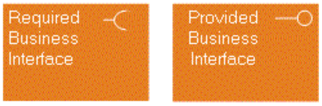
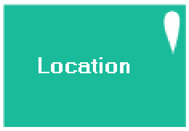
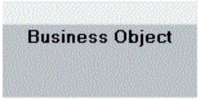
The ArchiMate language defines three main layers, based on specializations of the core concepts :

- The Business Layer offers products and services to external customers, which are realized in the organization by business processes performed by business actors.
- The Application Layer supports the business layer with application services which are realized by (software) applications.
- The Technology Layer offers infrastructure services (e.g., processing, storage, and communication services) needed to run applications, realized by computer and communication hardware and system software.

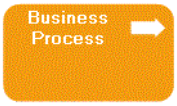
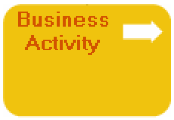
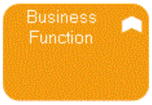
Archimate Business Layer

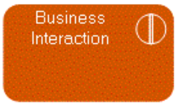
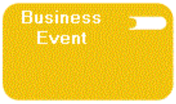
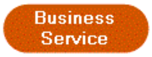
Structural concepts

Concept Name	Notation	Comment
Business Actor		A business actor is defined as an organizational entity capable of (actively) performing a behavior.
Business Role		A business role is defined as a named specific behavior of a business actor participating in a particular context
Business Collaboration		Business collaboration is defined as a (temporary) configuration of two or more business roles resulting in specific collective behavior in a particular context

Concept Name	Notation	Comment
Business Interface		A business interface declares how a business role can connect with its environment
Location		A Location is defined as a conceptual point or extent in space
Business Object		A business object is defined as a unit of information that has relevance from a business perspective

Behavioral concepts

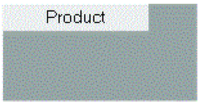
Concept Name	Notation	Comment
Business Process		A business process is defined as a unit of internal behavior or collection of causally-related units of internal behavior intended to produce a defined set of products and services.
Business Activity		A business activity is an elementary step of a business process. It expresses how a business activity contributes to the value chain represented by this process. It cannot be broken down.
Business Function		A business function is defined as a unit of internal behavior that groups behavior according to, for example, required skills, knowledge, resources, etc., and is performed by a single role within the organization.

Concept Name	Notation	Comment
Business Interaction		Business interaction is defined as a unit of behavior performed by a collaboration of two or more business roles.
Business Event		A business event is defined as something that happens (internally or externally) and influences behavior (business process, business function, business interaction)
Business Service		A business service is defined as the externally visible ("logical") functionality, which is meaningful to the environment and is realized by business behavior (business process, business function, or business interaction).

Informational concepts

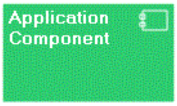
*Informational concepts are outside of the product and are not implemented in this version of **HOPEX Archimate**.*

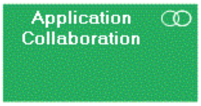
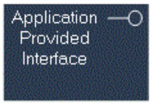
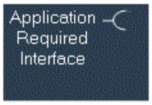
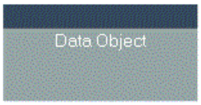
Concept Name	Notation	Comment
Representation	Not implemented	Representation is defined as the perceptible form of the information carried by a business object.
Meaning	Not implemented	Meaning is defined as the knowledge or expertise present in the representation of a business object, given a particular context.

Concept Name	Notation	Comment
Value	Not implemented	Value is defined as that which makes some party appreciate a service or product, possibly in relation to providing it, but more typically to acquiring it.
Product		A product is defined as a coherent collection of services, accompanied by a contract/set of agreements, which is offered as a whole to (internal or external) customers.
Contract	Not implemented	A contract is defined as a formal or informal specification of an agreement that specifies the rights and obligations associated with a product.

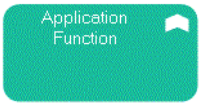
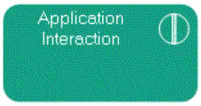
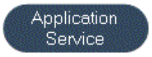
Archimate Application Layer

Structural concepts

Concept Name	Notation	Comment
Application Component		An application component is defined as a modular, deployable, and replaceable part of a system that encapsulates its contents and exposes its functionality through a set of interfaces.

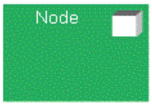
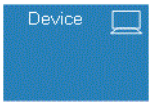
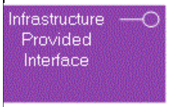
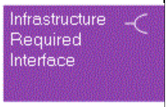
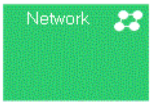
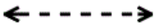
Concept Name	Notation	Comment
Application Collaboration		Application collaboration is defined as a (temporary) configuration of two or more components that co-operate to jointly perform application interactions
Application Interface	 	An application interface declares how an application component can connect with its environment.
Data Object		A data object is defined as a coherent, self-contained piece of information suitable for automated processing.

Behavioral concepts

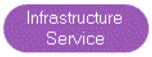
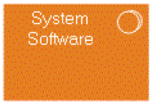
Concept Name	Notation	Comment
Application Function		An application function is defined as a representation of a coherent group of internal behavior of an application component.
Application Interaction		Application interaction is defined as a unit of behavior performed by a collaboration of two or more components.
Application Service		An application service is defined as an externally visible unit of functionality, provided by one or more components, exposed through well-defined interfaces, and meaningful to the environment.

Archimate Technology Layer

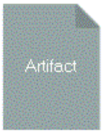
Structural concepts

Concept Name	Notation	Comment
Node		A node is defined as a computational resource upon which technical objects may be deployed for execution.
Device		A device is defined as a physical computational resource upon which technical objects may be deployed for execution.
Infrastructure Interface	 	An infrastructure interface is defined as a point of access where the functionality offered by a node can be accessed by other nodes and application components.
Network		A network is defined as a physical communication medium between two or more devices.
Communication Path		A communication path is defined as a link between two or more nodes, through which these nodes can exchange information.

Behavioral concepts

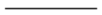
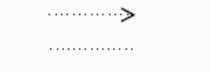
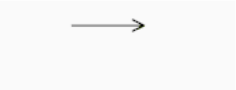
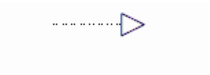
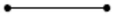
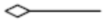
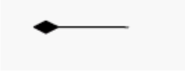
Concept Name	Notation	Comment
Infrastructure Service		An infrastructure service is defined as an externally visible unit of functionality, provided by one or more nodes, exposed through well-defined interfaces, and meaningful to the environment.
System Software		System software represents a software environment for specific types of components and objects that are deployed on it in the form of artifacts.

Informational concepts

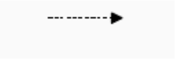
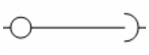
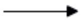
Concept Name	Notation	Comment
Artifact		An artifact is defined as a physical piece of information that is used or produced in a software development process, or by deployment and operation of a system.

Archimate RelationShips

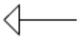
Structural RelationShips

Concept Name	Notation	Comment
Association		Association models a relationship between objects that is not covered by another, more specific relationship. note: this has only been implemented between data objects in the Information Structure Viewpoint.
Access		The access relationship models the access of behavioral concepts to business or data objects.
Used by		The used by relationship models the use of services by processes, functions, or interactions and the access to interfaces by roles, components, or collaborations.
Realization		The realization relationship links a logical entity with a more concrete entity that realizes it.
Assignment		The assignment relationship links units of behavior with active elements (e.g., roles, components) that perform them, or roles with actors that fulfill them.
Aggregation		The aggregation relationship indicates that an object groups a number of other objects.
Composition		The composition relationship indicates that an object consists of a number of other objects.

Dynamic RelationShips

Concept Name	Notation	Comment
Flow		The flow relationship describes the exchange or transfer of, for example, information or value between processes, function, interactions, and events
Interaction		The interaction relationship defines that two components are interactiong according to an interface protocol, using respective interfaces based on provided/ requireded services.
Triggering		The triggering relationship describes the temporal or causal relations between processes, functions, interactions, and events.

Other RelationShips

Concept Name	Notation	Comment
Junction		A junction is used to connect relationships of the same type. *Not implemented in this version of **HOPEX Archimate***
Specialization		The specialization relationship indicates that an object is a specialization of another object.

The ArchiMate Viewpoints

ArchiMate is composed of a set of diagrams, called viewpoints, which address different parts of an enterprise architecture.

Below are the definitions of these viewpoints as outlined in the ArchiMate Specification 2.1.

Introductory Viewpoint

The Introductory viewpoint forms a subset of the full ArchiMate language using a simplified notation. It is typically used at the start of a design trajectory, when not everything needs to be detailed yet, or to explain the essence of an architecture model to non-architects that require a simpler notation. Another use of this basic, less formal viewpoint is that it tries to avoid the impression that the architectural design is already fixed, an idea that may easily arise when using a more formal, highly structured or detailed visualization.

We use a simplified notation for the concepts, and for the relations. All relations except “triggering” and “realization” are denoted by simple lines; “realization” has an arrow in the direction of the realized service; “triggering” is also represented by an arrow. The concepts are denoted with slightly thicker lines and rounded corners, which give a less formal impression.

*Not implemented in this version of **HOPEX Archimate**.*

Organization Viewpoint

The Organization viewpoint focuses on the (internal) organization of a company, a department, a network of companies, or of another organizational entity (i.e a Business Actor). It is possible to present models in this viewpoint as organizational charts. The Organization viewpoint is very useful in identifying competencies, authority, and responsibilities in an organization.

Actor Co-operation Viewpoint

The Actor Co-operation viewpoint focuses on the relations of actors with each other within a Business Collaboration. It is very useful in determining external dependencies and collaborations and shows the value chain or network in which the actor operates.

Org Process Viewpoint

The Organizational Process viewpoint is used to describe the implementation of one business processes in a given organizational context. It emphasizes the breakdown of the described process into activities.

Next to the processes themselves, this viewpoint contains other directly related concepts, such as:

- The services a business process offers to the outside world, showing how a process contributes to the realization of the company’s products
- The realized business process
- The assignment of activities to Business actors
- The usage of business or application services by activities

The information (business objects via the intermediate object / work products concepts) used by the business process

Business Function Viewpoint

The Business Function viewpoint shows the main business functions of an organization and their relations in terms of the flows of information, value, or goods between them. Business functions are used to represent the most stable aspects of a company in terms of the primary activities it performs, regardless of organizational changes or technological developments. Therefore, the business function architecture of companies that operate in the same market often exhibit close similarities. The business function viewpoint thus provides high-level insight in the general operations of the company, and can be used to identify necessary competencies, or to structure an organization according to its main activities.

Business Process Viewpoint

The Business Process viewpoint is used to show the high-level structure and composition of one business process. Next to the processes themselves, this viewpoint contains other directly related concepts, such as:

- The services a business process offers to the outside world, showing how a process contributes to the realization of the company's products
- The assignment of business processes to roles, which gives insight into the responsibilities of the associated actors
- The information used by the business process

Business Process Co-operation Viewpoint

The Business Process Co-operation viewpoint is used to show the relations of one or more business processes with each other and/or with their environment. It can both be used to create a high-level design of business processes within their context and to provide an operational manager responsible for one or more such processes with insight into their dependencies.

Important aspects of business process co-operation are:

- Causal relations between the main business processes of the enterprise
- Mapping of business processes onto business functions
- Realization of services by business processes
- Use of shared data
- Execution of a business process by the same roles or actors

Each of these can be regarded as a "sub-view" of the business process co-operation view.

Not implemented in this version of ***HOPEX Archimate****.*

Product Viewpoint

The Product viewpoint depicts the value these products offer to the customers or other external parties involved and shows the composition of one or more products in terms of the constituting (business or application) services, and the associated contract(s) or other agreements. It may also be used to show the interfaces through which this product is offered. A Product viewpoint is typically used in product development to design a product by composing existing services or by identifying which new services have to be created for this product, given the value a customer expects from it. It may then serve as input for business process architects and others that need to design the processes and ICT realizing these products.

Application Behavior Viewpoint

The Application Behavior viewpoint describes an Application Function carried out by an application detailing the internal behavior of an application; e.g. how it realizes one or more application services. This viewpoint is useful in designing the main behavior of applications.

Application Co-operation Viewpoint

The Application Co-operation viewpoint describes the relations between applications components in terms of the information flows between them, or in terms of the services they offer and use within an Application Collaboration.

Application Structure Viewpoint

The Application Structure viewpoint shows the structure of one application component. This viewpoint is useful in designing or understanding the main structure of application components and the associated data usgae; e.g., to break down the structure of the system under construction, or to identify legacy application components that are suitable for migration/integration.

Application Usage Viewpoint

The Application Usage viewpoint describes how applications are used to support one Org Process, and how they are used by other applications. It can be used in designing an application by identifying the services needed by business processes and other applications, or in designing business processes by describing the services that are available. Furthermore, since it identifies the dependencies of Org Processes upon applications, it may be useful to operational managers responsible for these processes.

Node infrastructure Viewpoint

The Node infrastructure viewpoint contains the comuting and communication hardware infrastructure elements supporting the application layer, such as physical devices or networks.

Device infrastructure Viewpoint

The Device infrastructure viewpoint contains the software and hardware infrastructure elements within a physical device, such as component physical devices or system software (e.g., operating systems, databases, and middleware).

Infrastructure Usage Viewpoint

The Infrastructure Usage viewpoint shows how applications are supported by the software and hardware infrastructure: the infrastructure services are delivered by the devices; system software and networks are provided to the applications. This viewpoint plays an important role in the analysis of performance and scalability, since it relates the physical infrastructure to the logical world of applications. It is very useful in determining the performance and quality requirements on the infrastructure based on the demands of the various applications that use it.

*Not implemented in this version of **HOPEX Archimate**.*

Implementation and Deployment Viewpoint

The Implementation and Deployment viewpoint shows how one or more applications are realized on the infrastructure. This comprises the mapping of (logical) applications and components onto (physical) artifacts, such as Enterprise Java Beans, and the mapping of the information used by these applications and components onto the underlying storage infrastructure; e.g., database tables or other files. Deployment views play an important role in the analysis of performance and scalability, since they relate the physical infrastructure to the logical world of applications. In security and risk analysis, deployment views are used to identify, for example, critical dependencies and risks.

*Not implemented in this version of **HOPEX Archimate**.*

Information Structure Viewpoint

The Information Structure viewpoint is comparable to the traditional information models created in the development of almost any information system. It shows the structure of the information used in the enterprise or in a specific business process or application, in terms of data types or (object-oriented) class structures. Furthermore, it may show how the information at the business level (business objects) is represented at the application level in the form of the data structures used there (data objects), and how these are then mapped onto the underlying infrastructure; e.g., by means of a database schema (artifact).

Service Realization Viewpoint

The Service Realization viewpoint is used to show how one business service is realized by the underlying processes (and sometimes by application components). Thus, it forms the bridge between the business products viewpoint and the business process view. It provides a “view from the outside” on one or more business processes.

Layered Viewpoint

The Layered viewpoint pictures several layers and aspects of an enterprise architecture in one diagram. There are two categories of layers, namely dedicated layers and service layers. The layers are the result of the use of the “grouping” relation for a natural partitioning of the entire set of objects and relations that belong to a model. The infrastructure, the application, the process, and the actors/ roles layers belong to the first category. The structural principle behind a fully layered viewpoint is that each dedicated layer exposes, by means of the “realization” relation a layer of services, which are further on “used by” the next dedicated layer. Thus, we can easily separate the internal structure and organization of a dedicated layer from its externally observable behavior expressed as the service layer that the dedicated layer realizes. The order, number, or nature of these layers are not fixed, but in general a (more or less) complete and natural layering of an ArchiMate model will contain the succession of layers depicted in the example given below. However, this example is by no means intended to be prescriptive. The main goal of the Layered viewpoint is to provide overview in one diagram. Furthermore, this viewpoint can be used as support for impact of change analysis and performance analysis or for extending the service portfolio.

*Not implemented in this version of **HOPEX Archimate**.*

Landscape Map Viewpoint

A landscape map is a matrix that represents a three-dimensional coordinate system that represents architectural relations. The dimensions of the landscape maps can be freely chosen from the architecture that is being modeled. In practice, often dimensions are chosen from different architectural domains; for instance, business functions, application components, and products. Note that a landscape map uses the ArchiMate concepts, but not the standard notation of these concepts.

In most cases, the vertical axis represents behavior like business processes or functions; the horizontal axis represents “cases” for which those functions or processes must be executed, such as different products, services market segments, or scenarios; the third dimension represented by the cells of the matrix is used for assigning resources like information systems, infrastructure, or human resources. The value of cells can be visualized by means of colored rectangles with text labels. Obviously, landscape maps are a more powerful and expressive representation of relations than traditional cross tables. They provide a practical manner for the generation and publication of overview tables for managers, process, and system owners. Furthermore, architects may use landscape maps as a resource allocation instrument and as an analysis tool for the detection of patterns and changes in this allocation.

*Not implemented in this version of **HOPEX Archimate**.*

THE MEGA IMPLEMENTATION OF ARCHIMATE

As an Enterprise Architecture tool, the **HOPEX** Modeling application can be used to support an ArchiMate based EA initative on top of other frameworks and languages. Many different frameworks have been designed to help implement enterprise architecture projects. Even though common concepts can be retrieved in each framework, vocabulary may differ and sometimes the same words can be used with slight variations. **HOPEX** has made its own methodology choices and the vocabulary contained in the **HOPEX** Modeling tool metamodel reflects **HOPEX**'s EA vision to be methodologically consistent over various frameworks.

Because of these differences, it is important to first establish the common vocabulary that will be used in the tool. For the **HOPEX** implementation of ArchiMate, the choice was made to use the ArchiMate vocabulary for ArchiMate users while keeping a common metamodel shared by all users.

HOPEX Archimate relies on an "aliasing" mechanism to present the ArchiMate vocabulary to ArchiMate users. Alias of concepts are defined upon the **HOPEX** standard metamodel and can be changed using **HOPEX Studio**. See "Concepts mapping", page 84.

Archimate Diagramming Layers

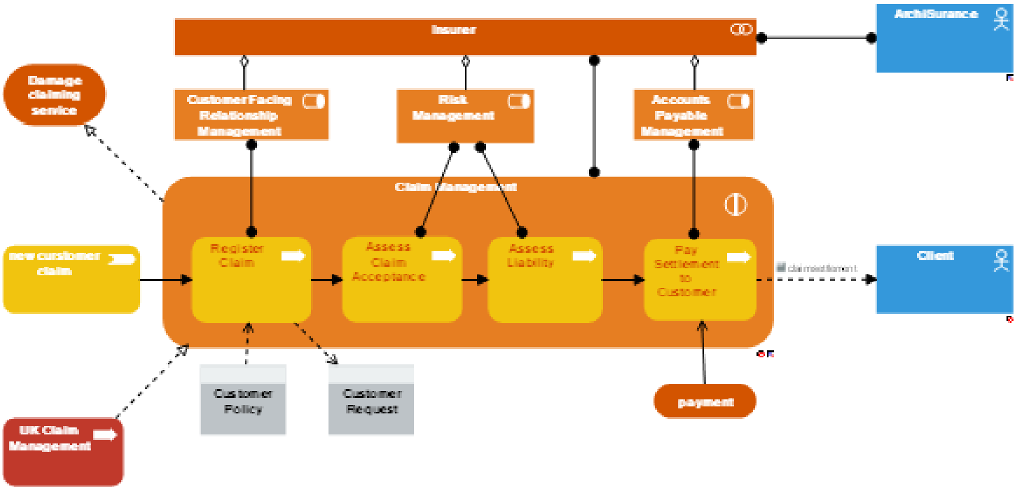
HOPEX Archimate aims at providing architects with the core content of the ArchiMate standard whilst ensuring compatibility with the other **HOPEX** products and solutions.

Each layer is covered by a set of viewpoints diagrams describing the main objects of the layer, and the associated reports.

Business Layer viewpoints

The Business Layer viewpoints cover the description of organizations and their responsibilities, business functions and processes which provide business services to customers or other processes;

Business Layer viewpoints are compatible with **HOPEX Business Process Analysis** models and **HOPEX Service Design** service definition models.



HOPEX Archimate Business Process Viewpoint

Described concept (ArchiMate)	Describing Diagram
Business Actor	Organization Viewpoint
Business Collaboration (Business Role of Business Collaboration type)	Actor Co-operation Viewpoint
Business Function (Business Process of Business Function type)	Business Function Viewpoint
Business Process	Business Process Viewpoint
Org Process	Organizational Process Viewpoint *Note : this is added to ArchiMate for consistency sake with **HOPEX** framework*
Product	Product Viewpoint

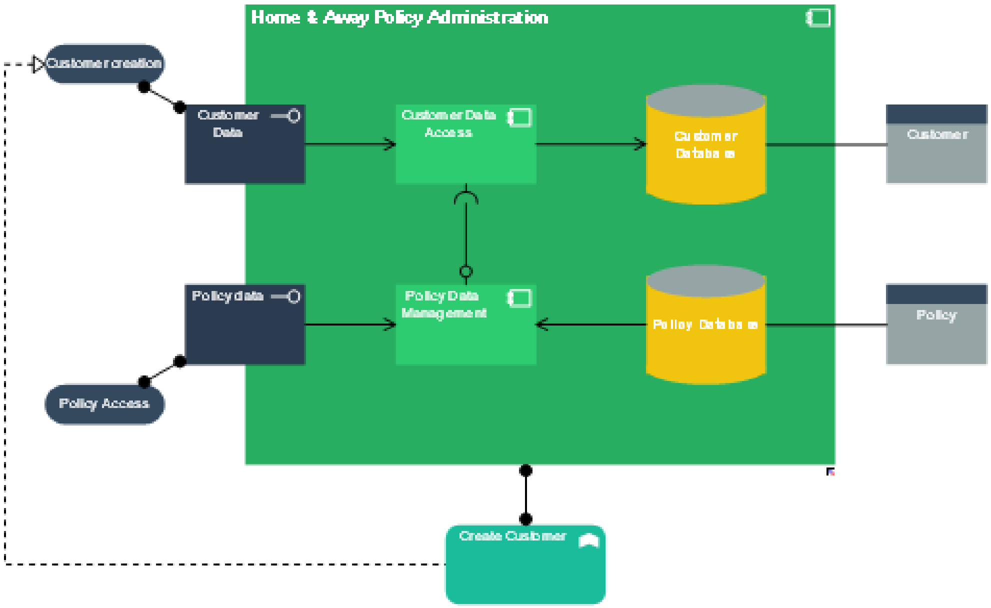
Application Layer viewpoints

The Application Layer viewpoints cover the description of application components and application functions which provide application services to the business processes.

Application Layer viewpoints are compatible with and **MEGA System Oriented IT Architecture** application models.

Information aspects are addressed relying on both **MEGA Information Architecture** (for the business objects) and **HOPEX Logical Data / HOPEX Database Builder** / **HOPEX UML** (for the data objects).

.



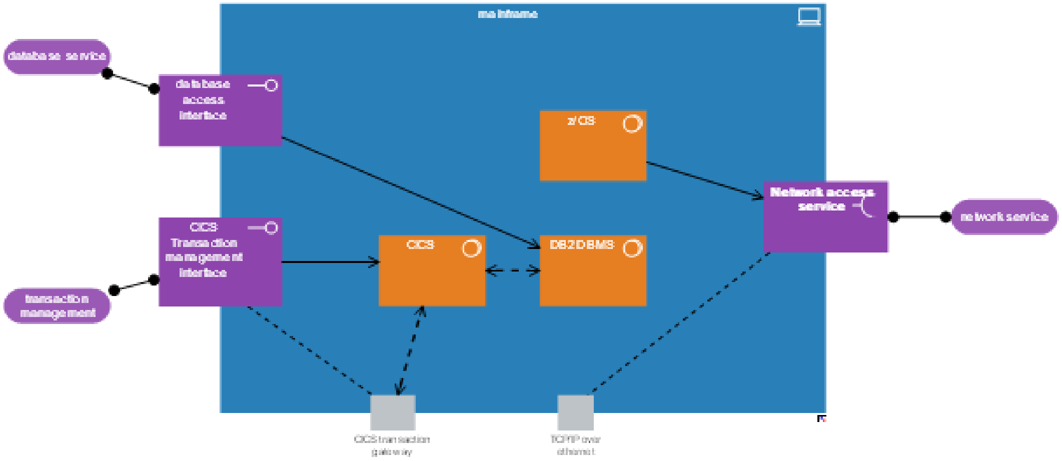
HOPEX Archimate Application Structure Viewpoint

Described concept (ArchiMate)	Describing Diagram
IT Function	Application Behavior Viewpoint
Application Collaboration	Application Co-operation Viewpoint
Application Component	Application Structure Viewpoint
Org Process	Application Usage Viewpoint

Technology Layer viewpoints

The Technology Layer viewpoints cover the description of computing & communication infrastructures supporting the application components.

Technology Layer viewpoints are compatible with **HOPEX IT Architecture** "resource architecture" models.



HOPEX Archimate Device Infrastructure Viewpoint

Described concept (ArchiMate)	Describing Diagram
Node	Node Infrastructure Viewpoint
Device	Device Infrastructure Viewpoint

Cross Layer viewpoints

The Cross Layer viewpoints address the transverse viewpoints not linked to a specific layer.

- Information Structure Viewpoint: enables the creation of a "data model" defining associations between data objects and linking data objects to business objects.
- Service Realization Viewpoint: enables the summary of the realization of a business service by Business Processes, Org Processes and Application Services and Components

Described concept (ArchiMate)	Describing Diagram
Information Model	Information Structure Viewpoint
(Business) Service	Service Realization Viewpoint

DISCOVERING THE DESKTOP

The **HOPEX Archimate** comes with ergonomic tools - such as the viewpoint oriented desktop - to help the ArchiMate expert to use the **HOPEX** Modeling tool. Before giving you step-by-step directions on how to create Viewpoints diagrams that describes the layers components of your architecture and how to produce the associated deliverables, this chapter starts by presenting the basic features of **HOPEX Archimate**.

HOPEX ARCHIMATE DESKTOP PRESENTATION

☛ ***HOPEX Archimate** is mainly intended for web users. Desktops described in this guide are accessible only to Web desktop users.*

Connecting to the Solution

To connect to **HOPEX Archimate**, see HOPEX Common Features, "HOPEX Web Front-End Desktop".

The HOPEX Archimate Desktop

The menus and commands available in **HOPEX Archimate** depend on the profile with which you are connected.

☛ *For more details on using the Web platform for HOPEX solutions, see the **HOPEX Common Features** guide.*

☛ *For more details on the method used to connect a user to a desktop, see "Link a user to an enterprise plan", page 36.*

Dynamic interface presentation

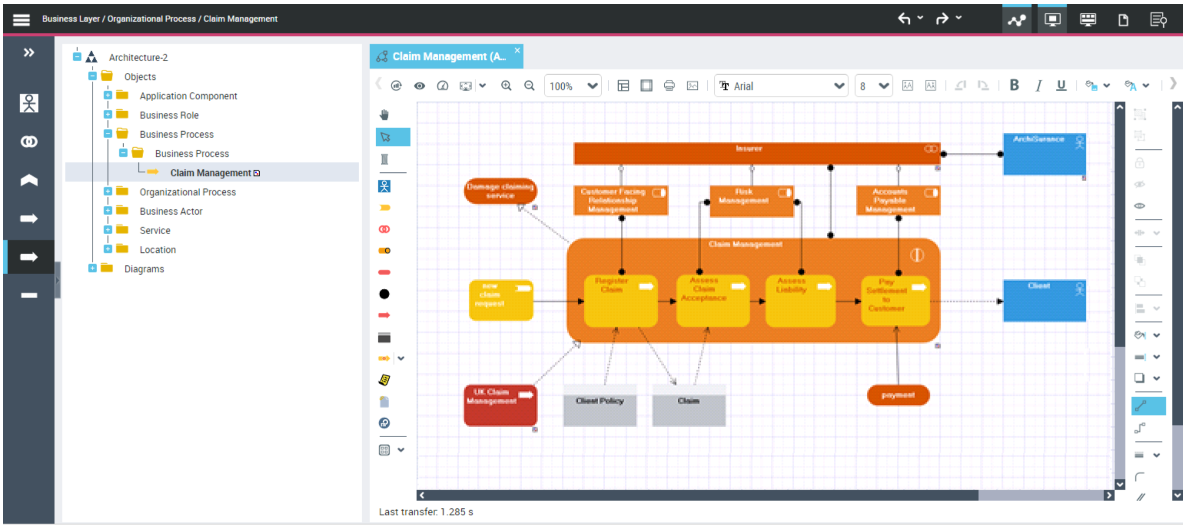
HOPEX Archimate provides "viewpoint" oriented desktop which has been specifically fine-tuned for ArchiMate.

☛ *The **HOPEX Archimate** is primarily intended for web client usage; the ArchiMate viewpoint desktop presented here is only available as a web desktop.*

The desktop entry point is a new "Architecture" concept which gathers objects of interest for a user or a set of users; an Architecture is created according to a Framework, and the Architecture elements can be browsed according to defined Viewpoints.

Creating an Architecture object according to the ArchiMate Framework is a pre requisite to using the viewpoint oriented desktop.

Each viewpoint is associated with diagram types, report types and metaclasses which are of interest for the viewpoint, enabling the user to easily access a given diagram and to easily enable drag and drop of relevant objects on this diagram.



HOPEX Archimate viewpoint web desktop

Presentation of space common to all roles

All users have access to the **HOPEX Archimate** desktop and access to the following panes:

- **Home** and **To-Do-List** that are common to all **HOPEX** solution users.
- **Reports**: accesses all reports, improving understanding of terms and their use.

Each Architecture View contains the same elements.

- An **Objects** folder, where the various objects relevant for a given viewpoint is entitled to; basically all the inventory objects candidate to be used in a view diagram (e.g. Business Actor can be reused across Business Process Viewpoint diagrams) should be available. Contextual objects (e.g. a "Business Activity" of a Business Process) are available only in the diagram toolbar
  - Main objects of a viewpoint may be described by a diagram which has been defined as creatable in a given viewpoint (e.g. "Business

Process" can be described by a "Business Process Viewpoint diagram" in a "Business Process viewpoint" view)
  - When an Object can be described by several diagrams (e.g. Org Process can be described by a "Org Process Viewpoint Diagram" and an "Application Usage Viewpoint Diagram")
  - If an object is already described by a diagram (e.g. a Business Actor by an "Organization Viewpoint diagram")
- A **My Objects** folder, where user-defined and view-specific sub folders can be created to gather several objects.
- A **Diagrams** folder, where all the diagrams of the allowed type for the view are available and may be opened (note that new diagrams can be created only from their described object).
- A **Documentation** folder, where all the generated reports of the allowed report template type in the currently browsed viewpoint can be found.

Presenting the Archimate Enterprise Architect space

The Archimate Enterprise Architect manages architecture projects. Can create ArchiMate architecture projects, and assign these to users: business architects and business specialists for example.

The Archimate Enterprise Architect is responsible for execution of the following tasks:

- ArchiMate Architectures creation.
- Assignment of participants to ArchiMate Architecture.

In addition to the panes offered in standard mode to all **HOPEX Archimate** desktop users, the Archimate Enterprise Architect has access to the following panes:

- the **Environment** pane extended to **Librairies** and **Tools** navigation menus,
- the **Business Layer** pane extended to **Organization**, **Actor Co-operation**, **Business Function**, **Business Process**, **Organization Process** and **Product** navigation menus,
- the **Application Layer** pane extended to **Application Behavior**, **Application Co-operation**, **Application Structure** and **Application Usage** navigation menus,
- the **Technology Layer** pane extended to **Node Infrastructure** and **Device Infrastructure** navigation menus,
- the **Cross Layer** pane extended to **Information Structure** and **Service Realization** navigation menus.

Presenting the Archimate Business Architect space

The Archimate Business Architect has only the **To-Do-List**, **Reports** and **Business Layer** tabs.

☛ *For more details on Archimate Business Architect responsibilities, see "Archimate Business Layer", page 15 and "HOPEX Archimate ViewPoints", page 45.*

The **Business Layer** pane provides access to the following navigation menus:

- **Organization**,
- **Actor Co-operation**,
- **Business Function**,
- **Business Process**,
- **Organization Process**,
- **Product**.

Presenting the Archimate Application Architect space

The Archimate Application Architect has only the **To-Do-List**, **Reports** and **Application Layer** tabs.

> ☛ *For more details on Archimate Application Architect responsibilities, see "Archimate Application Layer", page 18 and "HOPEX Archimate ViewPoints", page 45.*

The **Application Layer** pane provides access to the following menus.

- **Application Behavior**,
- **Application Co-operation**,
- **Application Structure**,
- **Application Usage**.

Presenting the Archimate Technology Architect space

The Archimate Technology Architect has only the **To-Do-List**, **Reports** and **Technology Layer** tabs.

> ☛ *For more details on Archimate Technology Architect responsibilities, see "Archimate Technology Layer", page 20 and "HOPEX Archimate ViewPoints", page 45.*

The **Technology Layer** pane provides access to the following menus.

- **Node Infrastructue**,
- **Device Infrastructue**.

Presenting the Archimate Functional Administrator space

The Archimate Functional Administrator has rights on all objects and Architectures.

Prepares the work environment and creates elements required for management of projects.

Manages:

- Architectures.
- Users, assignment of profiles and access rights to the different project steps.

In addition, the Archimate Functional Administrator profile, he has access to the **Administration** pane.

HOPEX Archimate Functional Administration

The Archimate Enterprise Architect is responsible for the following tasks:

- "Using Libraries", page 40;
- "Creating an Architecture", page 41;
- "Link a User to an Architecture", page 41.

Using Libraries

☛ *For more details on managing libraries, see the **HOPEX Common Features** "Libraries" guide.*

In the context of the **HOPEX Archimate** solution, a library can contain an architecture or represent an external environment with respect to the architecture under study.

Accessing the list of libraries

To access the list of libraries from the **Environment** navigation pane:

- Click **Libraries** in the navigation menu.
  The Architectures tree for the repository is displayed. The library tree appears underneath.

Creating a library

To access the list of libraries from the **Environment** navigation pane:

1. Click **Libraries** in the navigation menu.
   The library tree is displayed over the architecture tree.
2. Select **New > Library**.
   A **Libraries** creation dialog box opens.
3. Specify the the name of the library.
4. If appropriate, enter the name of the **Owner**.
5. Click **OK**.
   The library appears in the tree.

Defining a Default Library

When you begin working in a repository, you should define a default library.

To define a default library:

1. Select the library in which your are going to be working.
2. Right-click the library and select **Set as default**.
   This library appears in the **Default Library** folder.

Creating an Architecture

Creating an architecture

To access the desktop and create architecture elements, a new architecture based on the ArchiMate framework is required.

To create an architecture:

1. In the navigation area of the **HOPEX** Desktop, select the **Environment** navigation pane.
   The list of architectures appears in the edit area.
2. Select **New** > **Architecture**.
   The new architecture is created in the architectures tree.
3. Open the properties window and select **Characteristics** tab.
4. Modify the **Name** of your architecture.
5. Click **OK**.

Connect the architecture to a library

To connect the architecture to a library:

1. Open the architecture properties window that interests you.
2. From the **Owner** field, select the library to which you want to connect the architecture.
3. In the **Environment** navigation pane, select **Libraries**.
   The Architecture tree appears with the new architecture.

Link a User to an Architecture

Managing an architecture with the **HOPEX Archimate** solution includes different steps. In the solution interface, each step is associated with a navigation pane.

With **HOPEX Archimate**, the phases of an enterprise plan are described in a **work environment**.

To define the work phases of an enterprise plan to which a user has access, you must:

- specify the work environment of an architecture
- allocate the user to the steps of an architecture
  Thus, when connected to **HOPEX Archimate**, the user will only see the selected work environment steps.

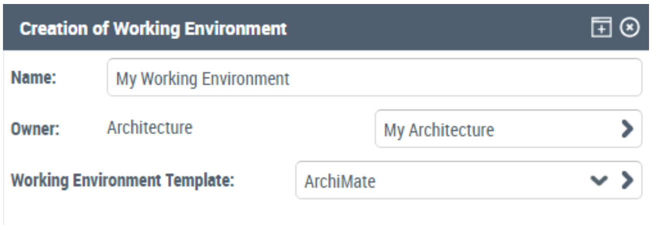
Specifying the work environment of an architecture

A user is allocated to an architecture through a **work environment**.

To link a new **work environment** to an architecture:

1. From the **Environment** navigation pane, select **Libraries** menu.and expand the **Architectures** tree.
2. Open the **Architecture** properties window that interests you.

3. Select the **Assignment** tab.
4. Click on the arrow to the left of the **Working environment** field and select **New**.
   A **Work environment** creation window opens.
5. Specify the name of the environment that you want to create, specify its **Archimate** type and click **OK**.



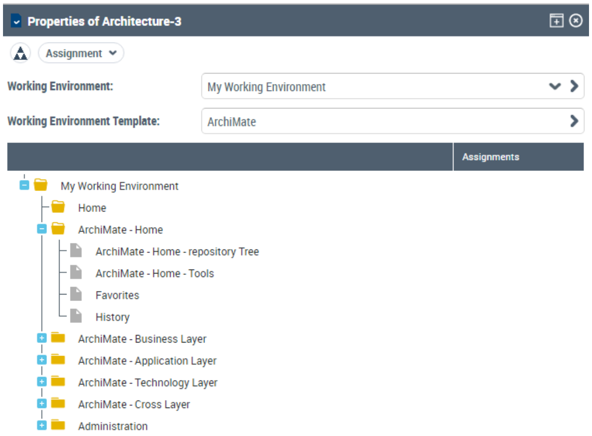
Creation of Working Environment

Name: My Working Environment

Owner: Architecture — My Architecture

Working Environment Template: ArchiMate

The tree with the steps of a **Archimate** type work environment appears on the bottom of the page.



Properties of Architecture-3

Working Environment: My Working Environment

Working Environment Template: ArchiMate

Allocating the user to the steps of an architecture

With the **HOPEX Archimate** solution, a user of the solution is a person (system).

To assign a user to all the steps of an architecture:

1. Open the Architecture properties window that interests you and select the **Assignments** tab.
2. At the tree root of the steps of the work environment, click the check box located under the **Assignments** column.
   An assignment window opens.
3. Click **Connect**.
   A dialog box for connecting persons (system) connection.
4. Find and select the persons (system) associated with the users that interest you.
5. Click **Add**.

6. In the assignment window, select the persons that you wish to connect to the enterprise plan and click **OK**.

   ☛ *In the same way, you can link a user to the steps of the work environment.*

7. Dispatch your private workspace so that the Architecture appears in the Architecture selection drop down list.

HOPEX ArchiMate ViewPoints

ArchiMate is composed of a set of viewpoints, which address different parts of an enterprise architecture. This chapter presents the functions offered by **HOPEX Archimate** to model the EA viewpoints of your enterprise through the diagrams proposed by ArchiMate to cover these viewpoints.

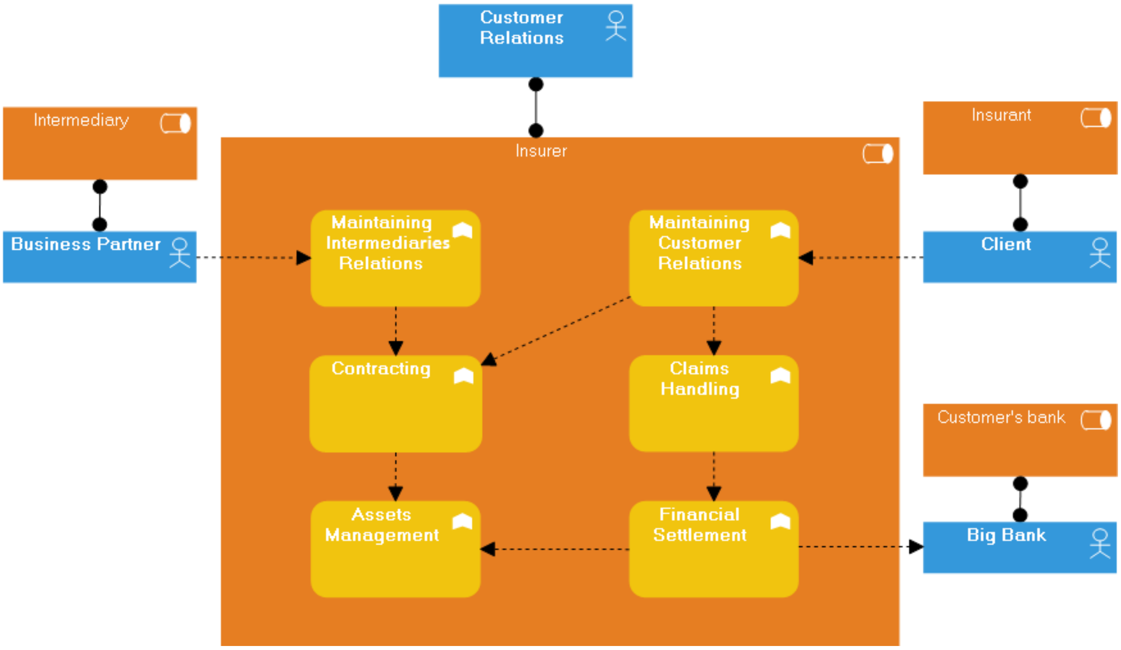
This presentation is based on the example of a fictional Insurance company which is used in the "ArchiMate 2.1 Specifiction", "ArchiMate Language Primer" and "ArchiSurance business case" documents.

Below are the definitions of these viewpoints as outlined in the ArchiMate Specification documents. The Organization ViewPoint paragraph will help you to start with **HOPEX Archimate**.

ARCHIMATE EXAMPLE CONTEXT PRESENTATION

"ArchiSurance" is a company that provides home and travel insurance services. The main business functions are :

- Maintaining Customer Relations and Intermediary Relations: these business functions are responsible for the contacts of the company with its customers and the intermediaries that sell its products. It handles customer questions and incoming claims, and performs marketing and sales.
- Contracting: this function handles the ‘back-office’ processing of contracts. It performs risk analysis and ensures legally and financially correct contracts.
- Claims Handling: this function is responsible for handling insurance claims.
- Financial Handling: this function performs the regular premium collection, according to the insurance policies with customers as produced by Contracting, and handles the payment of insurance claims.
- Asset Management: this function manages the financial assets of ArchiSurance, e.g. by investing in stocks and bonds.



Example of Business Function viewpoint diagram

ORGANIZATION VIEWPOINT

📖 *The Organization viewpoint focuses on the (internal) organization of a company, a department, a network of companies, or of another organizational entity (i.e. a Business Actor). It is possible to present models in this viewpoint as organizational charts. The Organization viewpoint is very useful in identifying competencies, authority, and responsibilities in an organization.*

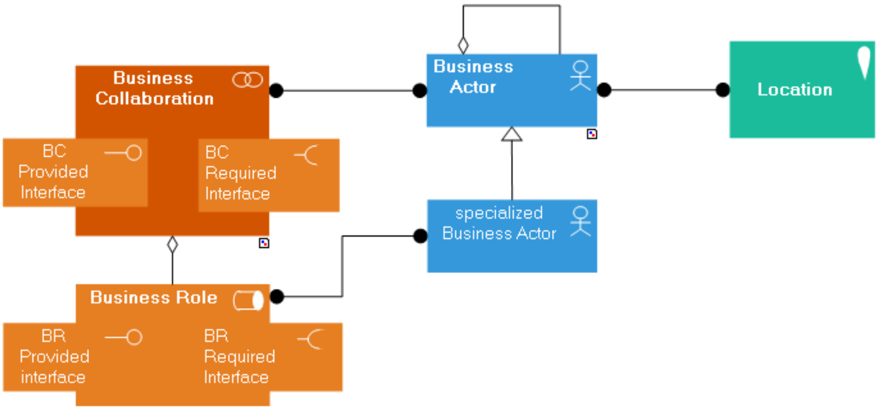
With this viewpoint, you will learn how to describe your first viewpoint diagram.

This presentation is based on the example of the "ArchiSurance" fictional insurance company, which is represented below.

Concepts and Relations

The ArchiMate concepts used in an Organization viewpoint Diagram and their relations are described.

This Viewpoint describes a Business Actor.



🕮 *A **Business Role** is defined as a named specific behavior of a **Business Actor** participating in a particular contex*

🕮 *A **Business Role** is assigned to a **Business Actor**.*

🕮 *A **Business Collaboration** is defined as a named specific behavior a (temporary) configuration of two or more business roles resulting in specific collective behavior in a particular context*

☛ *A **Business Collaboration** is a specialization of a **Business Role**. A **Business Collaboration** is an aggregation of several **Business Roles**.*

🕮 *A **Business Actor** is defined as an organizational entity capable of (actively) performing behavior.*

☛ *A **Business Role** uses the **Business Interfaces** which are associated with it.*

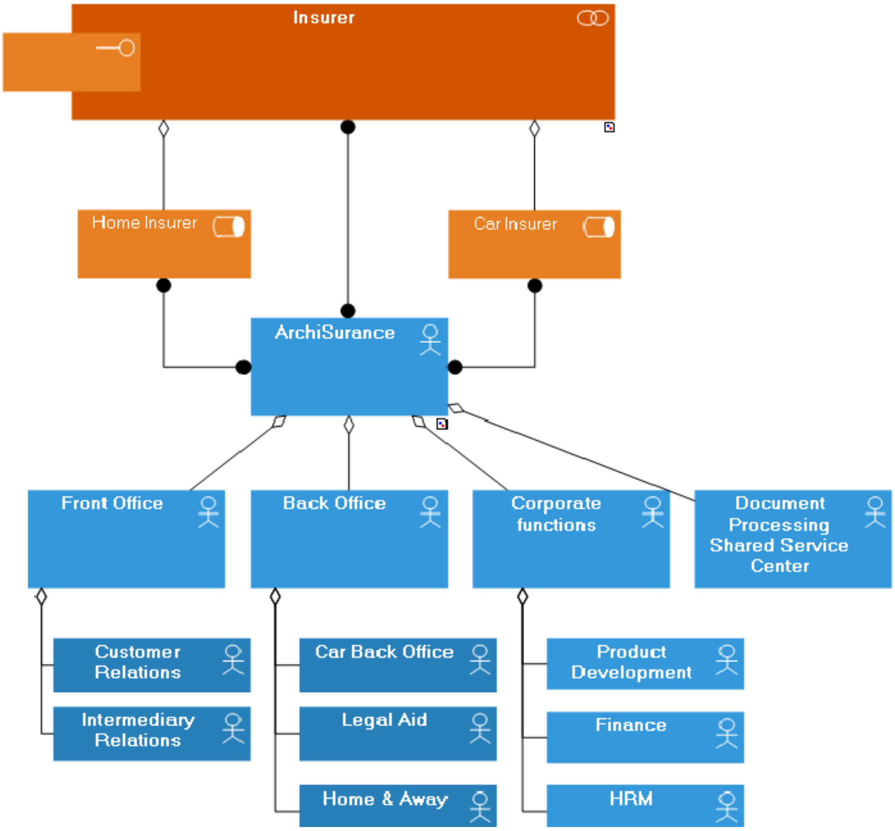
☛ *A **Business Interface** declares how a business role can connect with its environment*

Example Presentation

The diagram below describes the ArchiSurance organization.

The company is composed by :

- A front office, comprising departments for managing relations with customers on the one hand, and intermediaries on the other hand.
- Three separate back offices:
  - Home & Away: this department was the original pre-merger ArchiSurance, responsible for home and travel insurance.
  - Legal Aid: this is the old LegallyYours, responsible for legal aid and liability insurance.
  - Car: this department is the core of the old PRO-FIT and handles car insurance, including some legal aid.
- Furthermore, ArchiSurance is in the process of setting up a Shared Service Center for document processing, which will handle all document streams and performs scanning, printing, and archiving job.



Organization viewpoint diagram

Creating an Organizational Viewpoint Diagram

In **HOPEX Archimate**, the "organizational viewpoint" is a materialized by diagram describing a Business Actor.

Creating a diagram

To create an organizational viewpoint diagram:

1. In the **Business Layer** pane, select the **Organization** accordion item.
2. Unfold the **Objects** folder.
3. Right click on the **Business Actor** folder and select **New > Business Actor**.
   The **Add Business Actor** dialog box appears.
4. In the **Name** box, enter "ArchiSurance".
5. Right-click on the "ArchiSurance" Business Actor and select **New > Organizational ViewPoint**.

Creating Business Actors

You will define the Business Actors that defines the ArchiSurance organization structure

To create a Business Actor, for example the entity "Front Office".

1. In the insert toolbar, click the **Business Actor** button.
2. Click in the diagram.
   The **Add Business Actor** dialog box appears.
3. In the **Name** box, enter "Front Office"
4. Click **Create**.
   The actor appears in the diagram.

Creating several Business Actors

To create the other actors :

1. Double-click the **Business Actor** button and then click in the diagram
   The new actor appear in the diagram.
2. Press key <Esc> to stop the **Business Actors** creation.
3. To rename the actors, click the name of the actor, press key <F2> and enter a new name.

Assigning Sub-Actors to a Business Actor

To assign a Business Actor to another Business Actor, for example "Front Office" and "Back Office" to "ArchiSurance" actor:

1. Click on the **Link** button
2. Click the parent business actor and, holding down the mouse button, drag the cursor to the child business actor to be connected and release the mouse button.
   The two Business Actors are highlighted and a dotted line indicates the path that will be taken by the graphic link.

   ☛ *If several paths are possible: simply move the cursor to within the frame of the destination operation.*

Creating Business Roles

You will create Business Roles and connect each of these to the Business Actors responsible for their execution.

To create a Business Role participating, for example "Car Insurer".

1. In the insert toolbar, click the **Business Role** button.
2. Click in the diagram.
   The **Add Business Role** dialog box appears.

   > ☛ *If the business role that you wish to add already exists, click the arrow next to the "Name" field and find the business role that you are interested in (**Query > Find**).*

3. In the **Name** box, enter "Car Insurer".
4. Click **Create**.
   The role appears in the diagram.

Create the other business roles "Home Insurer" and "Insurant" in the same way.

To connect a business role to a business actor, use the same operating mode as before to connect business actor. An assignement relation link, connecting the nodes, appears in the diagram.

Creating Business Collaborations

Similarly create the **Business Collaborations**, as well as connection with dedicated **Business Roles**.

Creating a Business Interface

A Business Interface is used to describe the communication between two Business Roles. For example the Business Interface "Claim Request Handling" is used for communication between the "Insurant" and the "Insurer".

With **HOPEX Archimate**, a **Provided Business Interface** must be created for the "Insurer" and a **Required Business Interface** must be created for the "Insurant".

The process for creating a **Provided Business Interface** or **Required Business Interface** is identical.

To create a **Provided Business Interface** for the Business Collaboration "Insurer":

1. Select the Business Collaboration "Insurer" and open its Actor Co-operation viewpoint diagram,
2. In the insert toolbar of the diagram, click the **Provided Business Interface** button.
3. Click in the diagram.
   The **Adding a Provided Business Interface** dialog box appears.
4. Enter the **Name** of the new Interface and click **Add**.
   The **Creation of Provided Interface Protocol** dialog box appears.
5. Click the arrow at the right of the **Interface Protocol** field to define the interaction protocol enabling activation of this provided interface.
6. Select **Create Interface Protocol**.
   The **Creation Business Interface** dialog box appears.
7. Enter the Name of the new Interface Protocol. For example, "Claim Request Handling", and click **Ok**.
8. Click **OK** to close this dialog box.
   The Interface Protocol is automatically created.

9. Click **OK**.
   The Provided Interface appears in the diagram.

   ☛ *For more details, see "Using Interfaces and Interactions", page 79.*

Saving a Diagram

To save your drawing, click the **Save** button 💾.

ACTOR CO-OPERATION VIEWPOINT

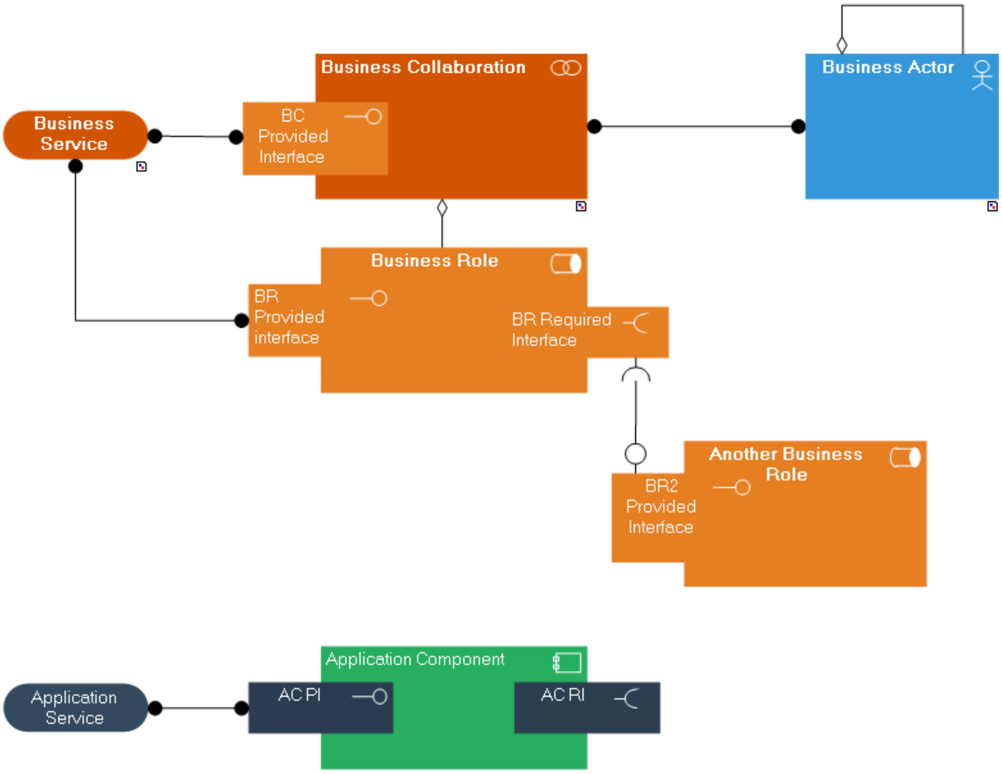
📖 *The Actor Co-operation viewpoint focuses on the relations of actors with each other within a Business Collaboration. It is very useful in determining external dependencies and collaborations and shows the value chain or network in which the actor operates.*

Concepts and Relations

An important use of the Actor Co-operation viewpoint is in showing how a number of cooperating business actors and/or application components together realize a business process.

The ArchiMate concepts used in an Actor Co-operation viewpoint Diagram and their relations are described below.

This viewpoint describes a Business Collaboration.



📖 *A* ***Business Service*** *is defined as the externally visible ("logical") functionality, which is meaningful to the environment and is realized by* *business behavior (Business Process, Business Function, or Business Interaction)*

☛ *In the context of an Actor Co-operation viewpoint, a **Business Service** is assigned to a **Business Interface***

📖 *An **Application Interface** declares how a component can connect with its environment.*

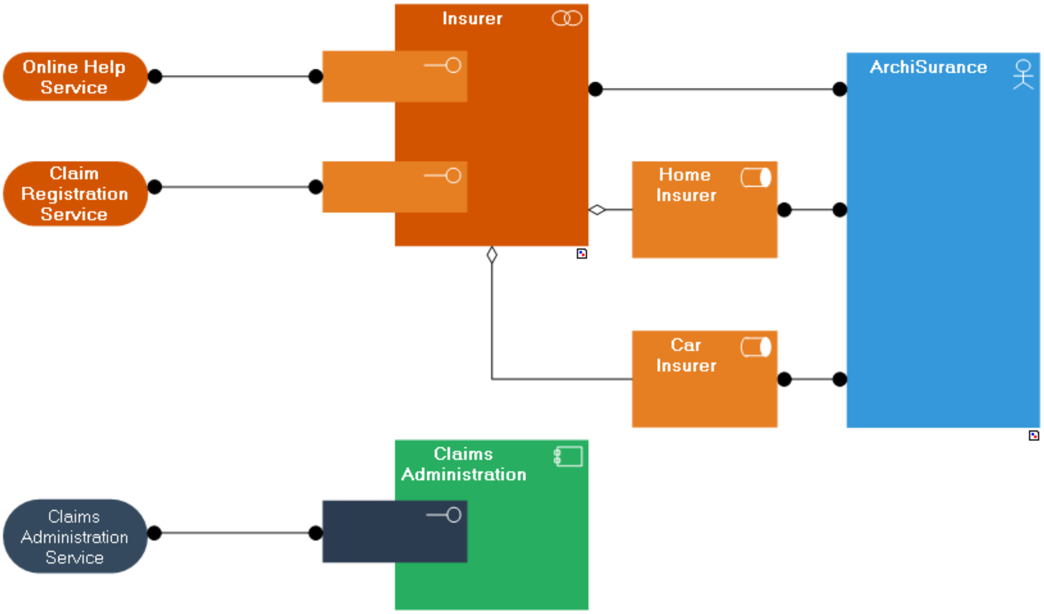
☛ *An **Application Service** may be assigned to an **Application Interface**.*

☛ *An **Application Interface** is owned by an **Application Component**. The composition link between the **Application Interface** and the **Application Component** indicates that an **Application Component** is a set of **Application Interfaces**.*

📖 *An **Application Component** is defined as a modular, deployable, and replaceable part of a system that encapsulates its contents and exposes its functionality through a set of interfaces.*

Example Presentation

The "Claim Registration Service" is available on the company portal. The Business service is assigned to an **Application Interface** noted "Insurance web services". This interface is used by the "Claim administration" application component and the corresponding application service is noted "Claim administration service".



ORG PROCESS VIEWPOINT

The Organizational Process viewpoint is used to describe the implementation of one business processes in a given organizational context. It emphasizes the breakdown of the described process into activities.

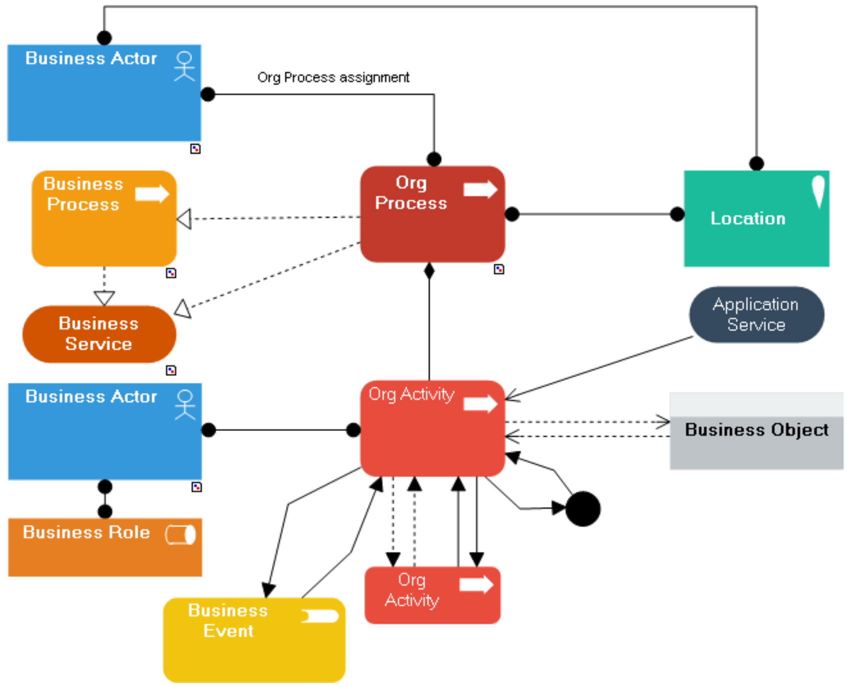
Next to the processes themselves, this viewpoint contains other directly related concepts, such as:

- The services a business process offers to the outside world, showing how a process contributes to the realization of the company's products
- The realized business process
- The assignment of activities to Business actors
- The usage of business or application services by activities

The information (business objects via the intermediate object / work products concepts) used by the business process

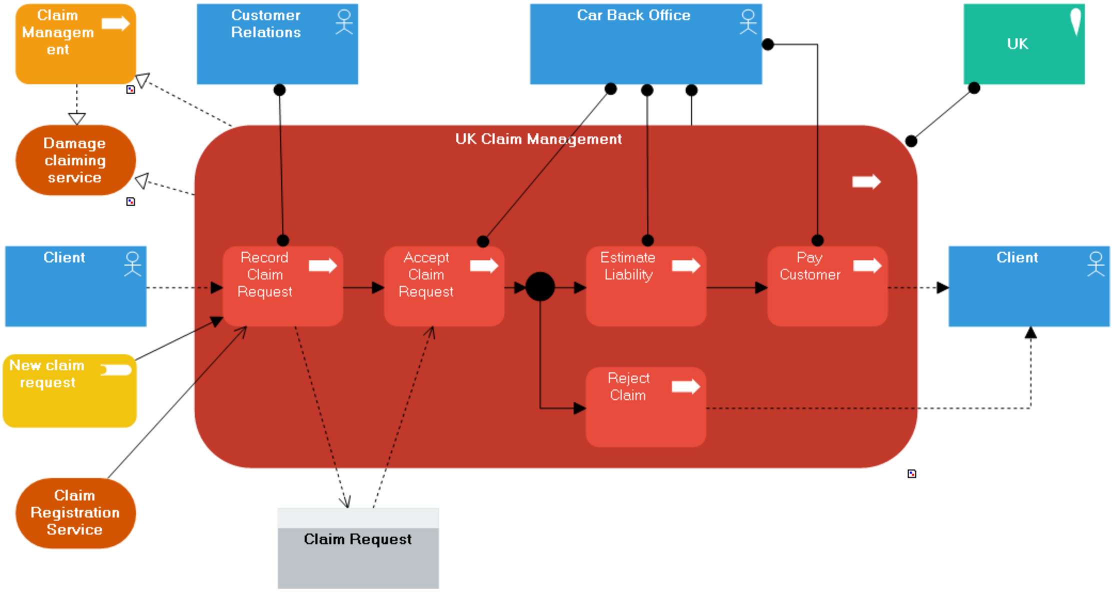
Concepts and Relations

The ArchiMate concepts used in a Organizational Process Viewpoint Diagram and their relations are described below. This viewpoint describes an Org Process



Example Presentation

The description of the Business Process implementation in a given conctext (here, the UK subsidiary) can be used to illustrate this viewpoint.



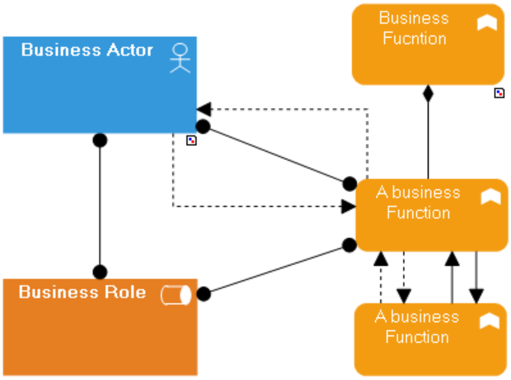
Example of Org Process viewpoint diagram

BUSINESS FUNCTION VIEWPOINT

📖 *The Business Function viewpoint shows the main business functions of an organization and their relations in terms of the flows of information, value, or goods between them. Business functions are used to represent the most stable aspects of a company in terms of the primary activities it performs, regardless of organizational changes or technological developments. Therefore, the business function architecture of companies that operate in the same market often exhibit close similarities. The business function viewpoint thus provides high-level insight in the general operations of the company, and can be used to identify necessary competencies, or to structure an organization according to its main activities.*

Concepts and Relations

The ArchiMate concepts used in a Business Function Viewpoint Diagram and their relations are described below. This viewpoint describes a Business Function.



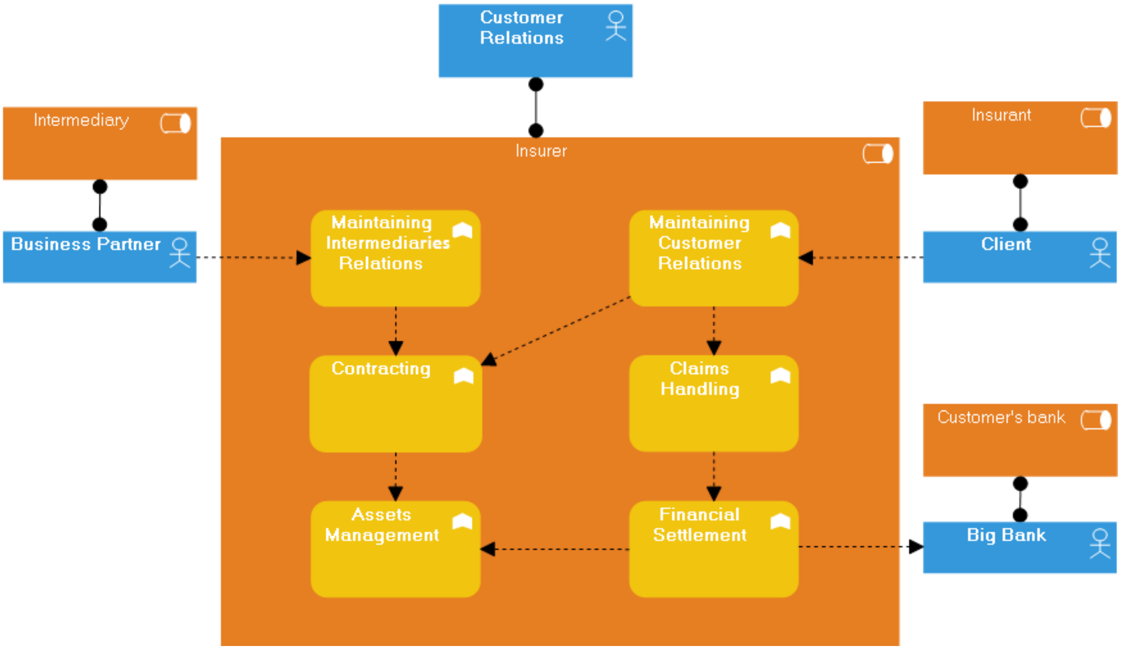
📖 *A **Business Role** is defined as a named specific behavior of a **Business Actor** participating in a particular contex*

📖 *A **Business Actor** is defined as an organizational entity capable of (actively) performing behavior.*

📖 *A **Business function** is defined as a unit of internal behavior that groups behavior according to, for example, required skills, knowledge, resources, etc., and is performed by a single role within the organization.*

Example Presentation

The description of the company functions can be used to illustrate this viewpoint.



Example of Business Function viewpoint diagram

BUSINESS PROCESS VIEWPOINT

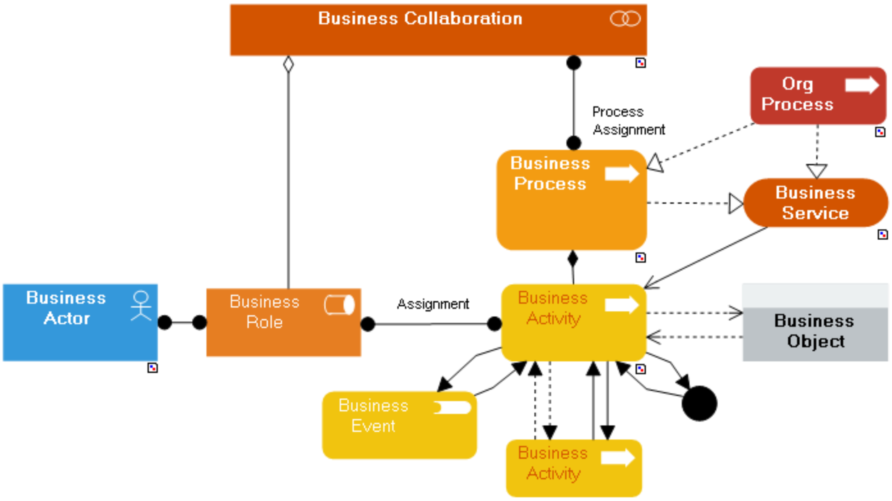
The Business Process viewpoint is used to show the high-level structure and composition of one business process. Next to the processes themselves, this viewpoint contains other directly related concepts, such as:

- The services a business process offers to the outside world, showing how a process contributes to the realization of the company’s products
- The assignment of business processes to roles, which gives insight into the responsibilities of the associated actors
- The information used by the business process

Concepts and Relations

The ArchiMate concepts used in a Business Process viewpoint Diagram and their relations are described below.

This viewpoint describes a Business Process or a Business Interaction



📖 *A **Business Collaboration** is defined as a named specific behavior a (temporary) configuration of two or more business roles resulting in specific collective behavior in a particular context*

📖 *A **Business Interface** declares how a business role can connect with its environment*

📖 *A **Business Role** is defined as a named specific behavior of a **Business Actor** participating in a particular contex*

📖 *A **Business Actor** is defined as an organizational entity capable of (actively) performing behavior.*

A ***Business process*** *is defined as a unit of internal behavior or collection of causally-related units of internal behavior intended to produce a defined set of products and services.*

☛ *A Business Activity is a step within a Business Process*

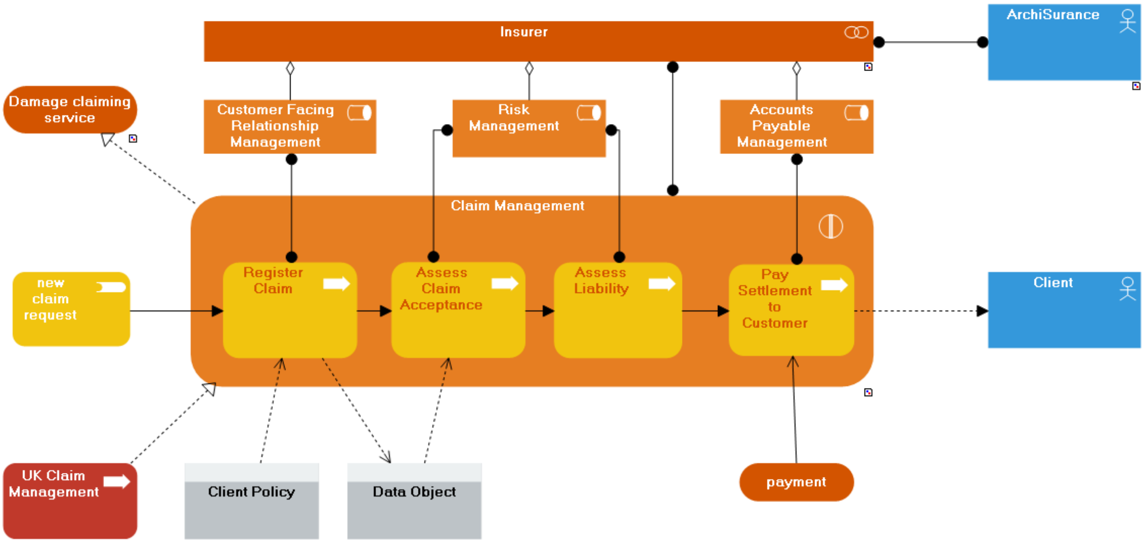
A ***Business event*** *is defined as something that happens (internally or externally) and influences behavior (business process, business function, business interaction).*

A ***Business object*** *is defined as a unit of information that has relevance from a business perspective.*

Representation *is not implemented in this version of* ***HOPEX Archimate****.*

Example Presentation

This diagram represents the functional description of "Damage Claiming process".



Example of Business Process viewpoint diagram

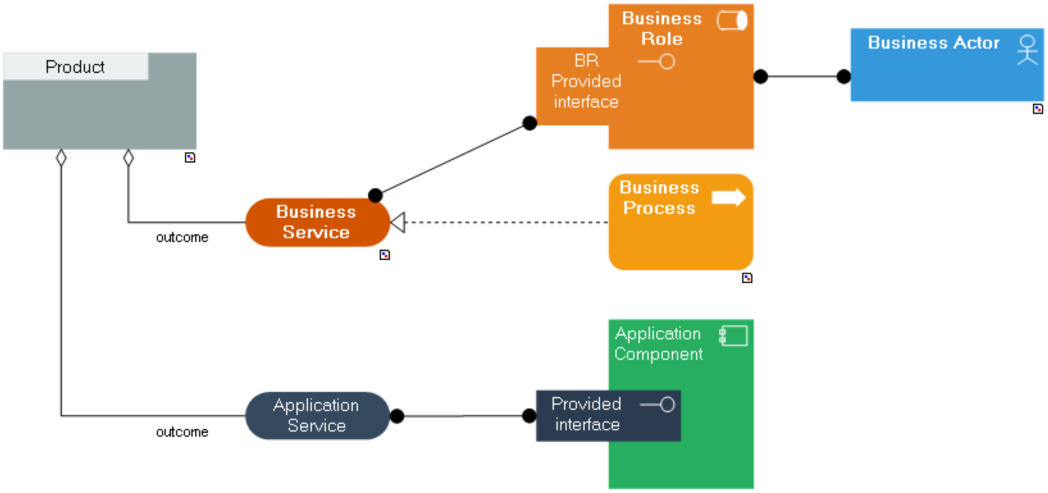
PRODUCT VIEWPOINT

📖 *The Product viewpoint depicts the value these products offer to the customers or other external parties involved and shows the composition of one or more products in terms of the constituting (business or application) services, and the associated contract(s) or other agreements. It may also be used to show the interfaces through which this product is offered. A Product viewpoint is typically used in product development to design a product by composing existing services or by identifying which new services have to be created for this product, given the value a customer expects from it. It may then serve as input for business process architects and others that need to design the processes and ICT realizing these products.*

Concepts and Relations

The ArchiMate concepts used in a Product viewpoint Diagram and their relations are described below.

This viewpoint describes a Product.



📖 *A **Business Interface** declares how a business role can connect with its environment*

📖 *A **Business Role** is defined as a named specific behavior of a **Business Actor** participating in a particular contex*

📖 *A **Business Actor** is defined as an organizational entity capable of (actively) performing behavior.*

📖 *A **Business process** is defined as a unit of internal behavior or collection of causally-related units of internal behavior intended to produce a defined set of products and services.*

📖 *A **Business service** is defined as the externally visible ("logical") functionality, which is meaningful to the environment and is realized by* *business behavior (business process, business function, or business interaction).*

📖 *An **Application Service** is defined as the externally visible ("logical") functionality, which is meaningful to the environment and is realized by business behavior (Business Process, Business Function, or Business Interaction)*

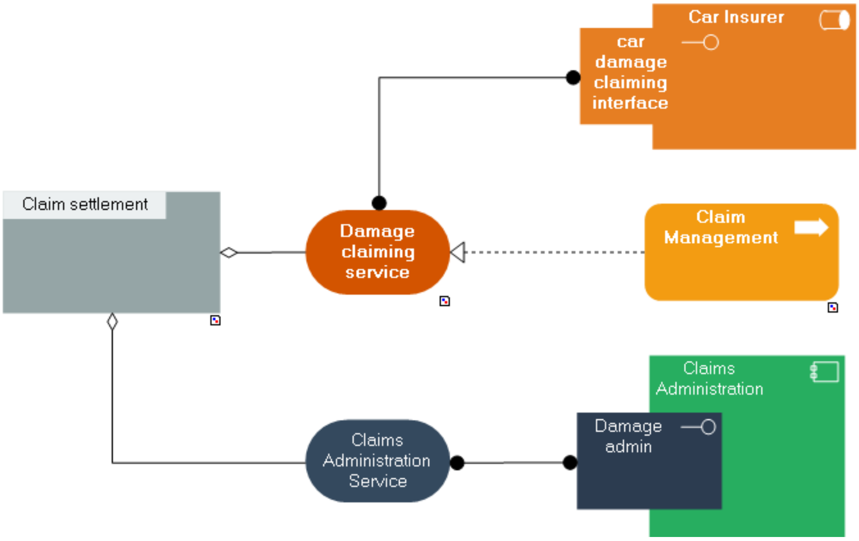
📖 *An **Application Interface** declares how a component can connect with its environment.*

📖 *An **Application Component** is defined as a modular, deployable, and replaceable part of a system that encapsulates its contents and exposes its functionality through a set of interfaces.*

__Value__, __Contract__ are not implemented in this version of __HOPEX Archimate__.

Example Presentation

This diagram shows how application services are contected to business processes, on the one hand, and realized by application components, on the other hand.



Example of Product viewpoint diagram

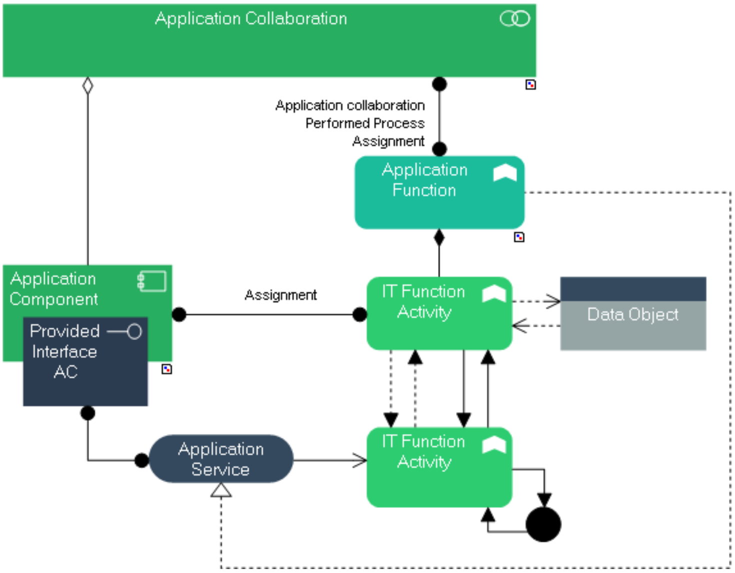
APPLICATION BEHAVIOR VIEWPOINT

📖 *The Application Behavior viewpoint describes an Application Function carried out by an application detailing the internal behavior of an application; e.g. how it realizes one or more application services. This viewpoint is useful in designing the main behavior of applications.*

Concepts and Relations

The ArchiMate concepts used in an Application Behavior viewpoint Diagram and their relations are described below.

This viewpoint describes an Application Function or an Application Interaction (subtypes of IT Function).



📖 *An **Application Collaboration** is defined as a (temporary) configuration of two or more components that co-operate to jointly perform application interactions*

📖 *An **Application Service** is defined as the externally visible ("logical") functionality, which is meaningful to the environment and is realized by business behavior (Business Process, Business Function, or Business Interaction)*

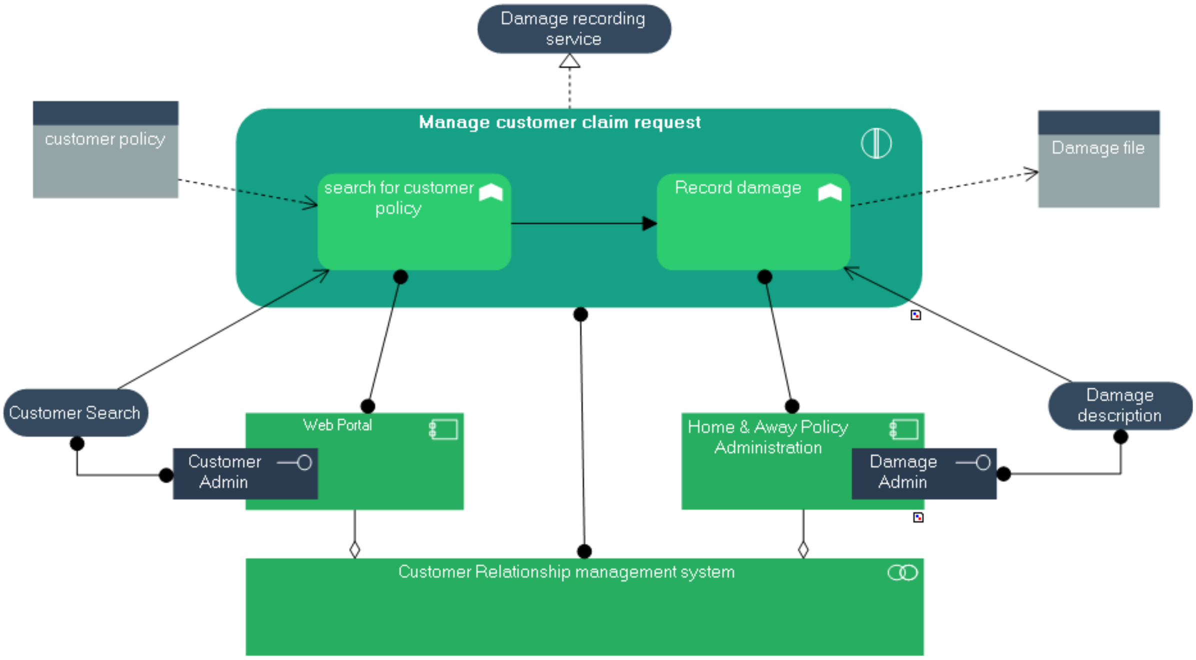
📖 *An **Application Interface** declares how a component can connect with its environment.*

📖 *An **Application Component** is defined as a modular, deployable, and replaceable part of a system that encapsulates its contents and exposes its functionality through a set of interfaces.*

📖 *An **Application Function** is defined as a representation of a coherent group of internal behavior of an application component.*

📖 *An **Data object** is defined as a coherent, self-contained piece of information suitable for automated processing.*

Example Presentation



Example of Application Behavior viewpoint diagram

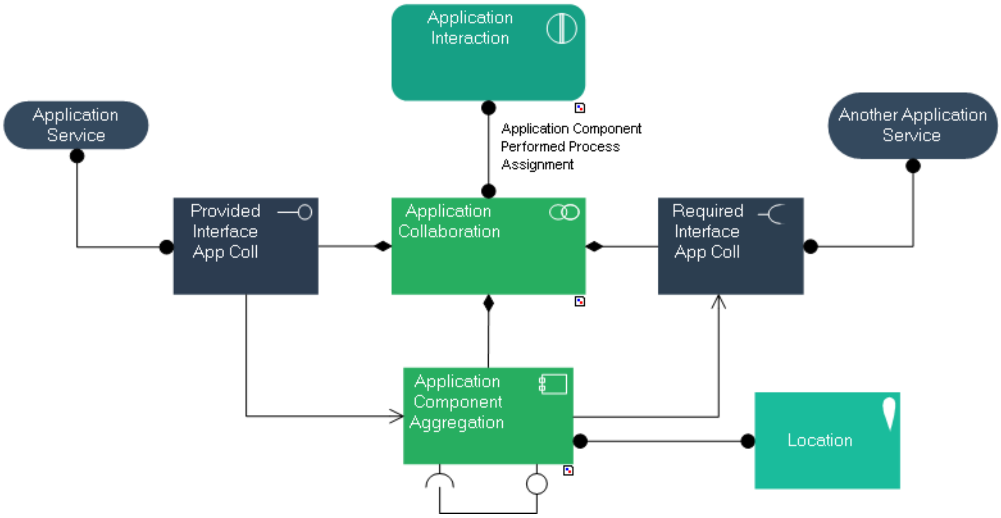
Application Co-operation Viewpoint

The Application Co-operation viewpoint describes the relations between applications components in terms of the information flows between them, or in terms of the services they offer and use within an Application Collaboration.

Concepts and Relations

The ArchiMate concepts used in an Application Co-operation viewpoint Diagram and their relations are described below.

This viewpoint describes an Application Collaboration.


Application Interaction
Application Component Performed Process Assignment
Application Service
Another Application Service
Provided Interface App Coll
Application Collaboration
Required Interface App Coll
Application Component Aggregation
Location

*An **Application Collaboration** is defined as a (temporary) configuration of two or more components that co-operate to jointly perform application interactions*

*An **Application Service** is defined as the externally visible ("logical") functionality, which is meaningful to the environment and is realized by business behavior (Business Process, Business Function, or Business Interaction)*

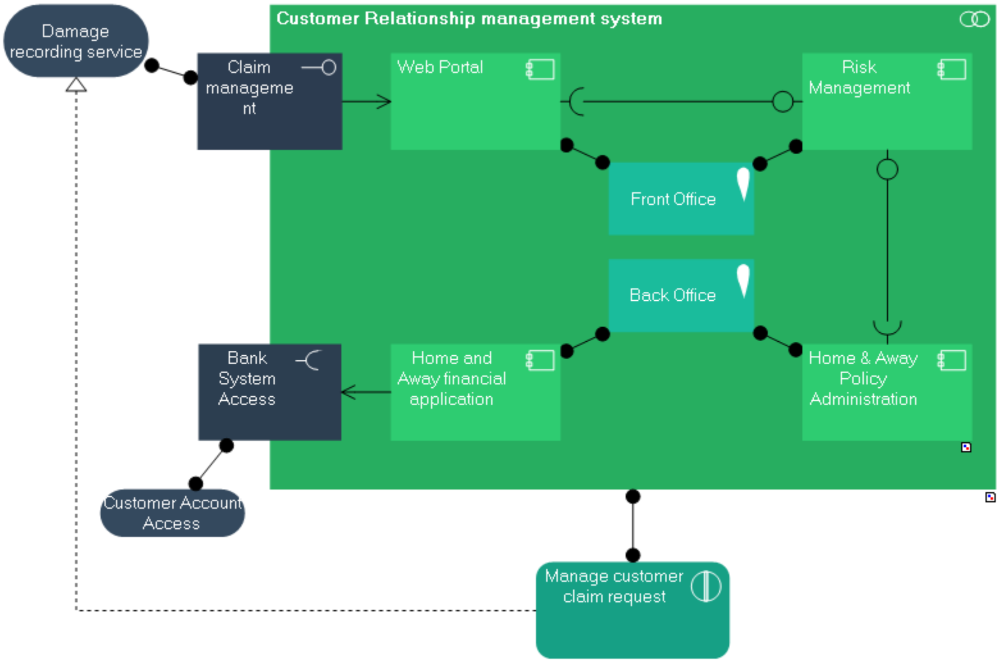
*An **Application Interface** declares how a component can connect with its environment.*

*An **Application Component** is defined as a modular, deployable, and replaceable part of a system that encapsulates its contents and exposes its functionality through a set of interfaces.*

*An **Application Interaction** is defined as a unit of behavior, in termes of information exchanges, performed by a collaboration of two or more components.*

Example Presentation

This example details the Customer Relationship Management Application collaboration, detailing how the involved Application Components interacts with each other so as to implement the Application interaction and provide the Application Services.


Customer Relationship management system
Damage recording service
Claim management
Web Portal
Risk Management
Front Office
Back Office
Bank System Access
Home and Away financial application
Home & Away Policy Administration
Customer Account Access
Manage customer claim request

Example of Application Co-operation viewpoint diagram

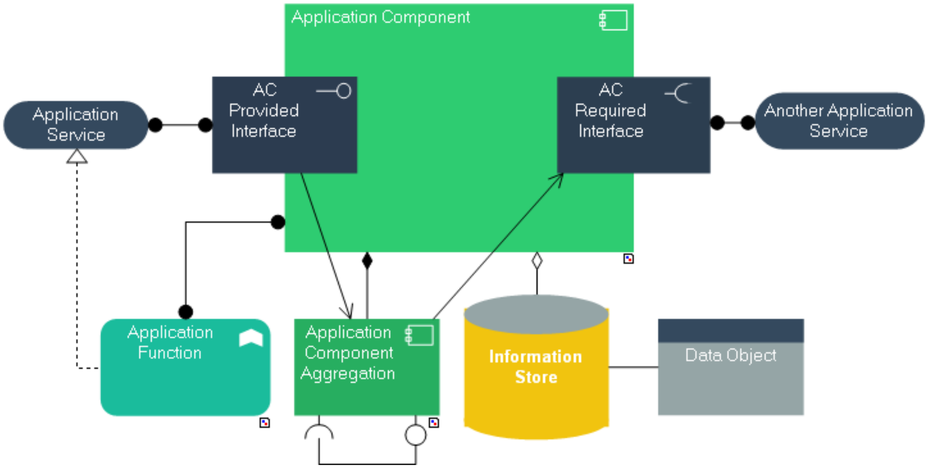
Application Structure Viewpoint

📖 *The Application Structure viewpoint shows the structure of one application component. This viewpoint is useful in designing or understanding the main structure of application components and the associated data usgae; e.g., to break down the structure of the system under construction, or to identify legacy application components that are suitable for migration/integration.*

Concepts and Relations

The ArchiMate concepts used in an Application Structure viewpoint Diagram and their relations are described below.

This viewpoint describes an Application Component.



📖 *An **Application Collaboration** is defined as a (temporary) configuration of two or more components that co-operate to jointly perform application interactions*

📖 *An **Application Interface** declares how a component can connect with its environment.*

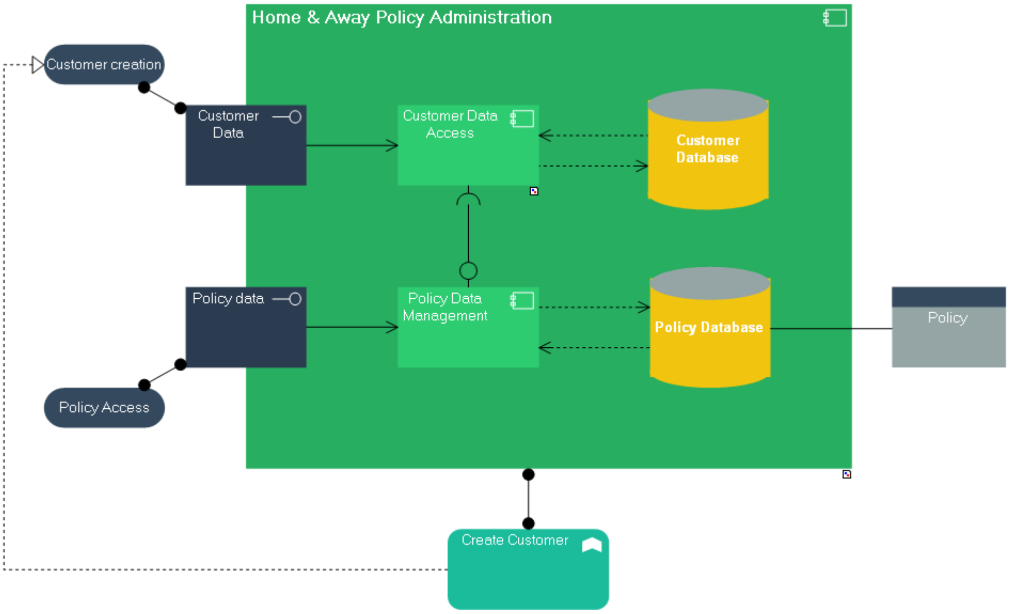
📖 *An **Application Interaction** is defined as a unit of behavior, in termes of information exchanges, performed by a collaboration of two or more components.*

📖 *An **Application Component** is defined as a modular, deployable, and replaceable part of a system that encapsulates its contents and exposes its functionality through a set of interfaces.*

☛ *An Information Store is the usage a database in the context of an Application Component; It can be accessed in Read mode or Write mode by an Application Component and hosts one ore several Information Models which contain the data objects (note: links between Data Object and Information Store are read only and can be displayed only if a link* *database > information model > data object has been created when defining the Information Structure Viewpoint).*

📖 *An **Data object** is defined as a coherent, self-contained piece of information suitable for automated processing.*

Example Presentation



Example of Application Structure viewpoint diagram

*The derived relationship between an Application Component and a data object is not implemented in this version of **HOPEX Archimate**.*

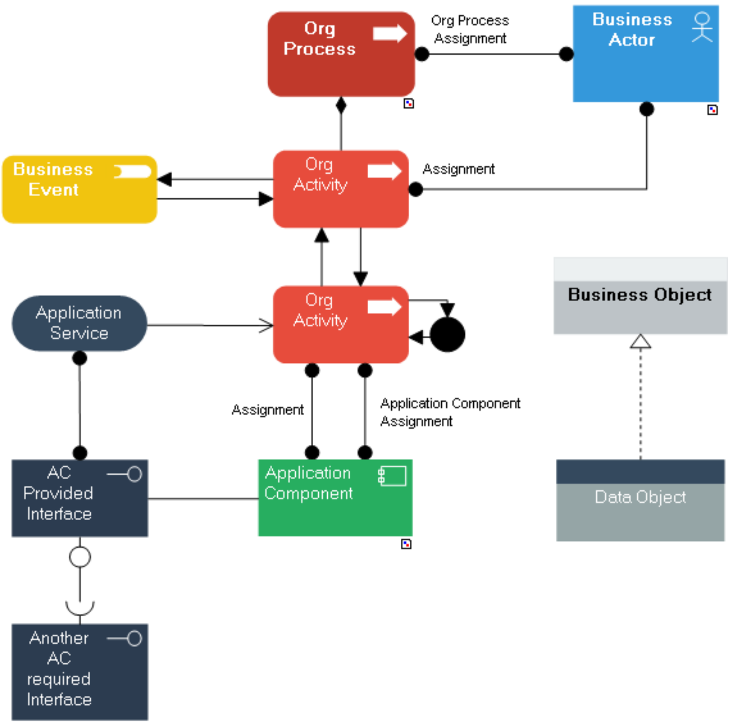
APPLICATION USAGE VIEWPOINT

📖 *The Application Usage viewpoint describes how applications are used to support one Org Process, and how they are used by other applications. It can be used in designing an application by identifying the services needed by business processes and other applications, or in designing business processes by describing the services that are available. Furthermore, since it identifies the dependencies of Org Processes upon applications, it may be useful to operational managers responsible for these processes.*

Concepts and Relations

The ArchiMate concepts used in an Application Usage viewpoint Diagram and their relations are described below.

This viewpoint describes an Org Process.



📖 *A* ***Business Role*** *is defined as a named specific behavior of a* ***Business Actor*** *participating in a particular contex*

📖 *A* ***Business object*** *is defined as a unit of information that has relevance from a business perspective.*

📖 *An* ***Application Service*** *is defined as the externally visible (“logical”) functionality, which is meaningful to the environment and is* *realized by business behavior (Business Process, Business Function, or Business Interaction)*

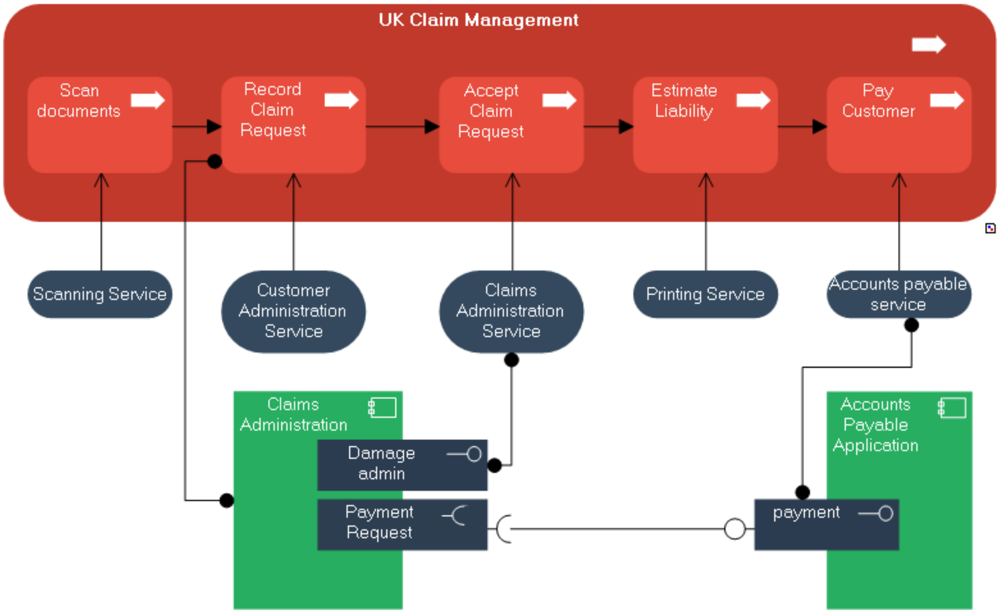
📖 *An **Application Interface** declares how a component can connect with its environment.*

📖 *An **Application Interaction** is defined as a unit of behavior, in termes of information exchanges, performed by a collaboration of two or more components.*

📖 *An **Application Component** is defined as a modular, deployable, and replaceable part of a system that encapsulates its contents and exposes its functionality through a set of interfaces.*

📖 *An **Data object** is defined as a coherent, self-contained piece of information suitable for automated processing.*

Example Presentation



Example of Application Usage viewpoint diagram

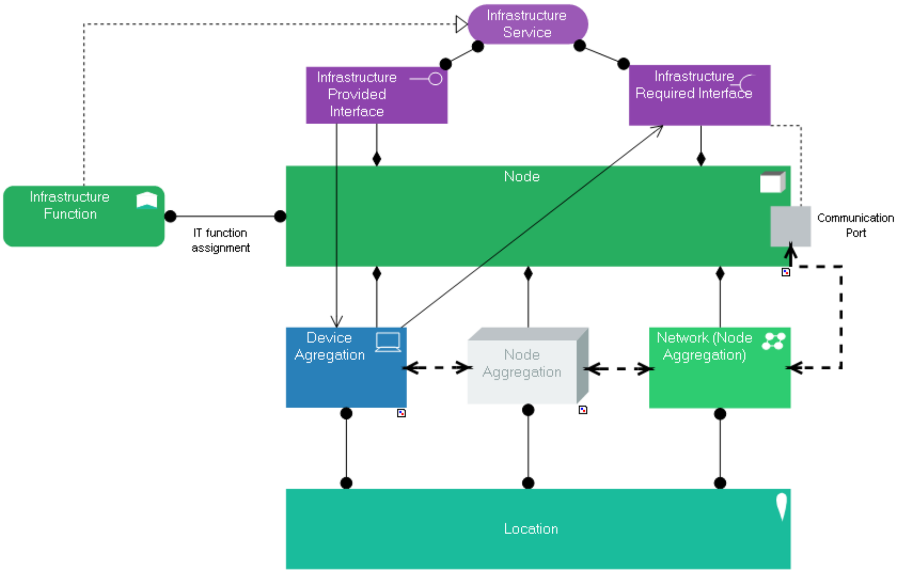
Node Infrastructure Viewpoint

📖 *The Node infrastructure viewpoint contains the computing and communication hardware infrastructure elements supporting the application layer, such as physical devices or networks.*

Concepts and Relations

The ArchiMate concepts used in a Node infrastructure viewpoint Diagram and their relations are described below.

This viewpoint describes a Node.



📖 *A* ***Node*** is defined as a computational resource upon which artifacts may be deployed for execution.

📖 *A* ***Device*** *is defined as a physical computational resource upon which artifacts may be deployed for execution.*

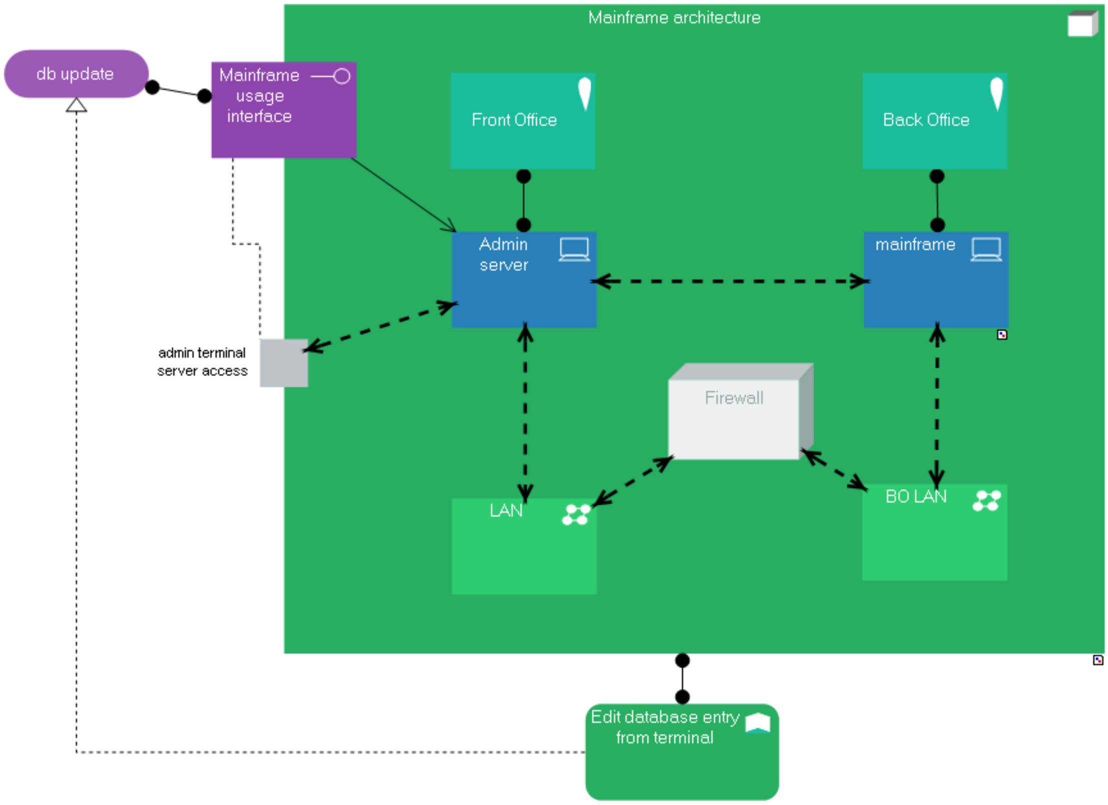
📖 *An* ***Infrastructure interface*** *is defined as a point of access where the functionality offered by a node can be accessed by other nodes and application components.*

📖 *A* ***Network*** *is defined as a physical communication medium between two or more devices.*

📖 *A* ***Communication path*** *is defined as a link between two or more nodes, through which these nodes can exchange information.*

📖 *An* ***Infrastructure service*** *is defined as an externally visible unit of functionality, provided by one or more nodes, exposed through well-defined interfaces, and meaningful to the environment.*

Example Presentation



Example of Node infrastructure viewpoint diagram

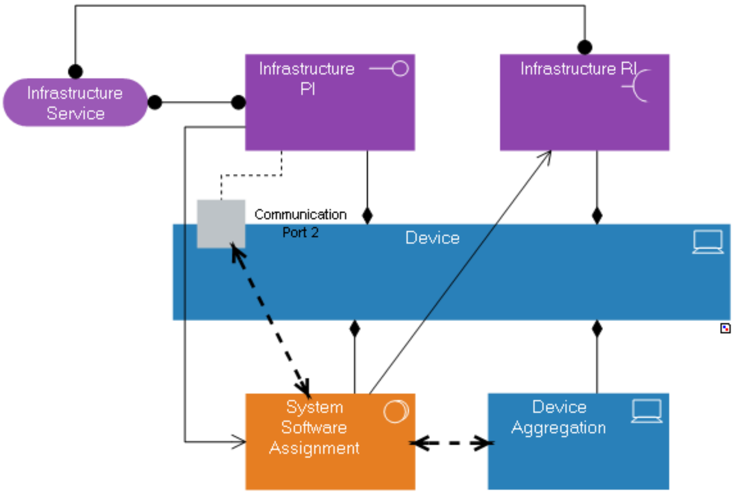
DEVICE INFRASTRUCTURE VIEWPOINT

📖 *The Device infrastructure viewpoint contains the software and hardware infrastructure elements within a physical device, such as component physical devices or system software (e.g., operating systems, databases, and middleware).*

Concepts and Relations

The ArchiMate concepts used in a Device infrastructure viewpoint diagram and their relations are described below.

This viewpoint describes a Device.



📖 *A **Device** is defined as a physical computational resource upon which artifacts may be deployed for execution.*

📖 *An **Infrastructure interface** is defined as a point of access where the functionality offered by a node can be accessed by other nodes and application components.*

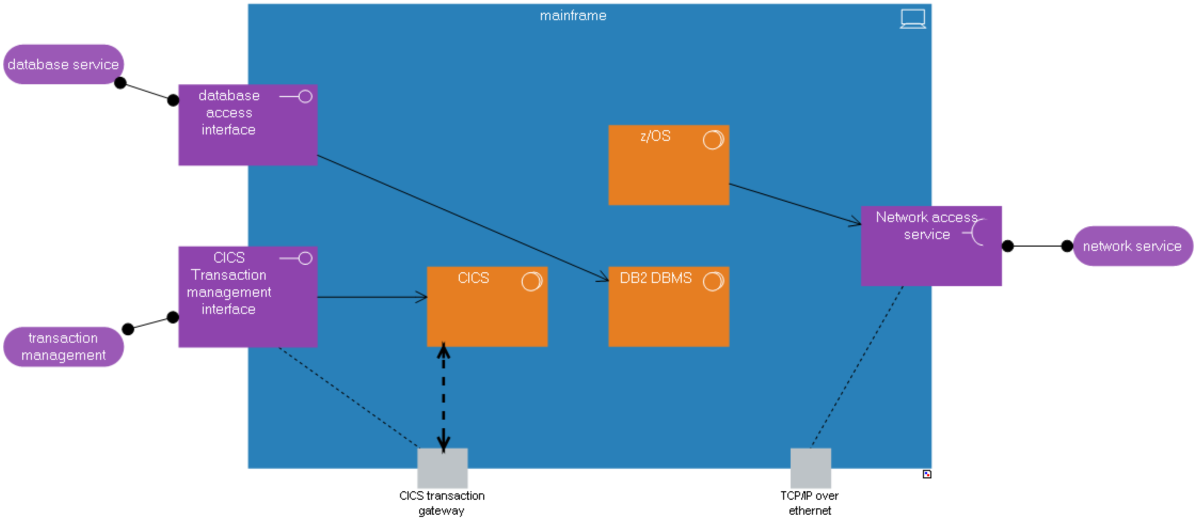
📖 *A **Communication path** is defined as a link between two or more nodes, through which these nodes can exchange information.*

📖 *An **Infrastructure service** is defined as an externally visible unit of functionality, provided by one or more nodes, exposed through well-defined interfaces, and meaningful to the environment.*

📖 *A **System software** represents a software environment for specific types of components and objects that are deployed using artifacts*

📖 *An **Artifact** is defined as a physical piece of information that is used or produced in a software development process, or by deployment and operation of a system.*

Example Presentation



Example of Device infrastructure viewpoint diagram

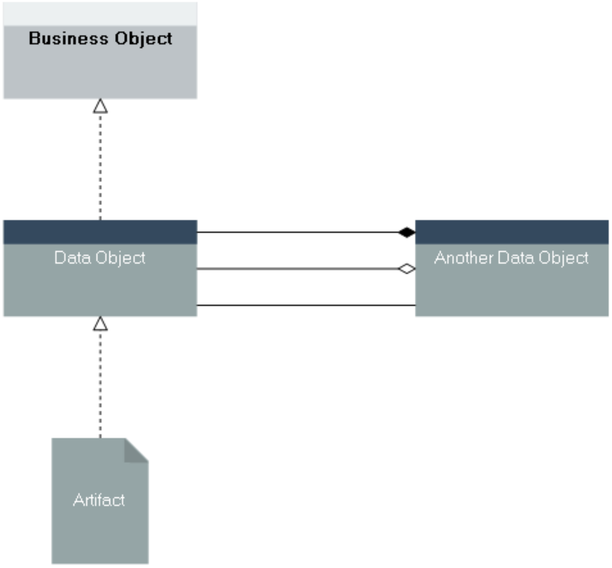
INFORMATION STRUCTURE VIEWPOINT

The Information Structure viewpoint is comparable to the traditional information models created in the development of almost any information system. It shows the structure of the information used in the enterprise or in a specific business process or application, in terms of data types or (object-oriented) class structures. Furthermore, it may show how the information at the business level (business objects) is represented at the application level in the form of the data structures used there (data objects), and how these are then mapped onto the underlying infrastructure; e.g., by means of a database schema (artifact).

Concepts and Relations

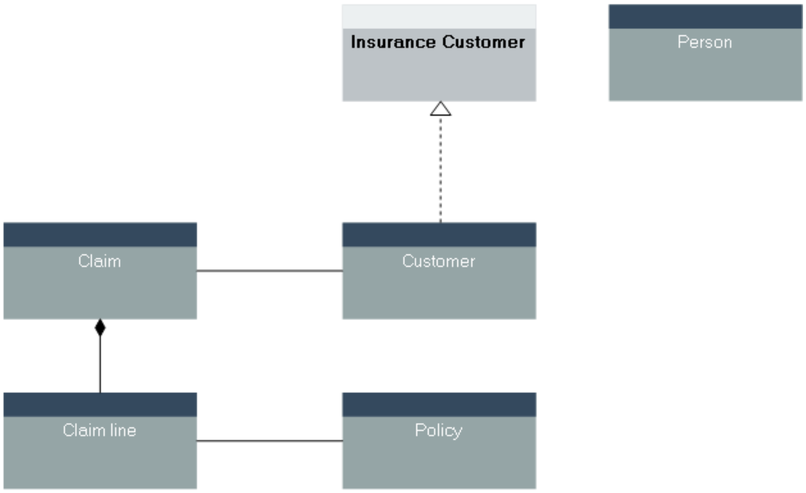
The ArchiMate concepts used in an Information Structure viewpoint Diagram and their relations are described below.

This viewpoint describes an Information Model which can describe the content of an Information Store.



*In this version of **HOPEX Archimate Meaning** and **Representation** are not implemented.*

Example Presentation



Example of Information Structure viewpoint diagram

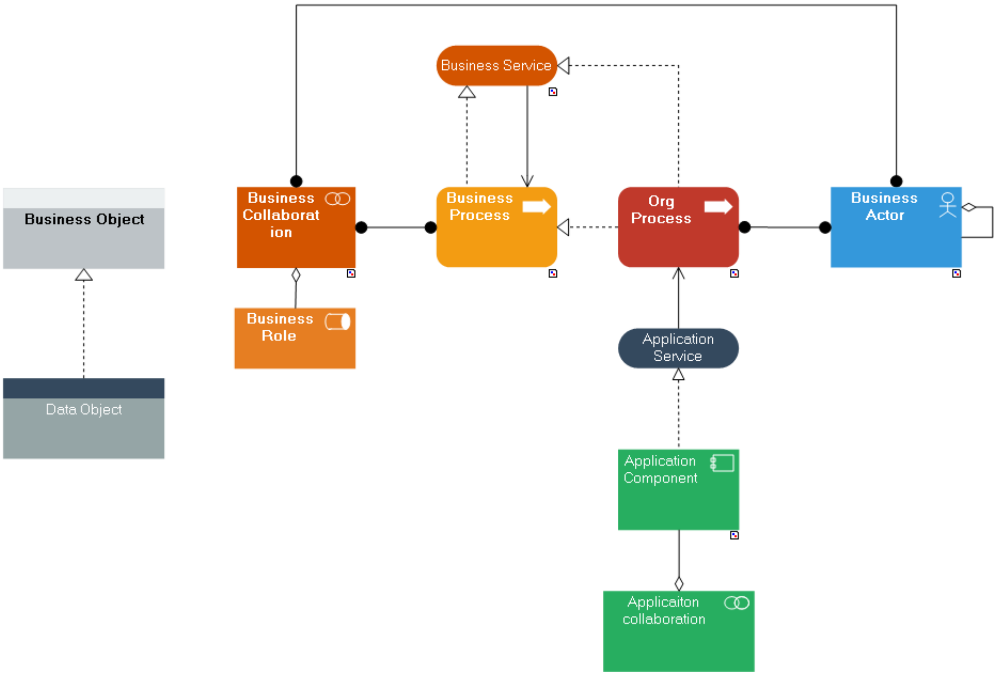
Service Realization Viewpoint

The Service Realization viewpoint is used to show how one business service is realized by the underlying processes (and sometimes by application components). Thus, it forms the bridge between the business products viewpoint and the business process view. It provides a "view from the outside" on one or more business processes.

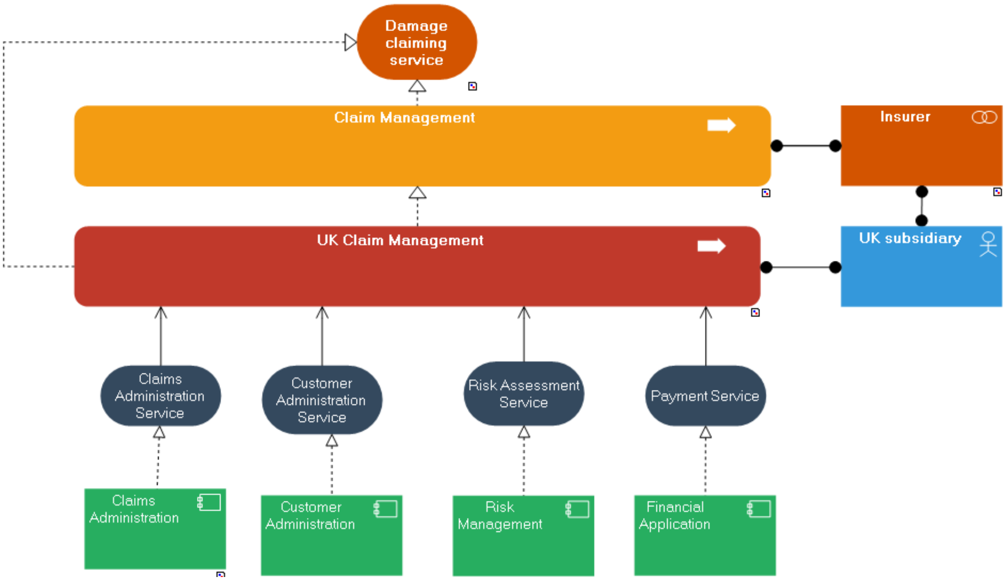
Concepts and Relations

The ArchiMate concepts used in a Service Realization viewpoint Diagram and their relations are described below.

This viewpoint describes a business service.



Example Presentation



Example of Service Realization viewpoint diagram

Using Interfaces and Interactions

A Business Interface is used to describe the communication between two Business Roles. For example the Business Interface "Claim Request Handling" is used for communication between the Insurant and the Insurer.

An Application Interface is used to describe the communication between two Application Components. For example the Application Interface "Web Claim Request Handling" is used to manage the web communication between the Insurant and the Insurer regarding the claim request over web.

An Infrastructure Interface is used to describe the communication between two Infrastructure Nodes or devices.

Introduction to Concepts

In **HOPEX Archimate**, a Business Collaboration or a Business Role, an Application Collaboration or an Application Component, a Device or a Node is created to assure one or several services.

These services are exposed through **Provided Interfaces** and used through **Required Interfaces**.

📖 A ***Provided Interface*** is a point of exchange by which an agent offers a service to potential customers.

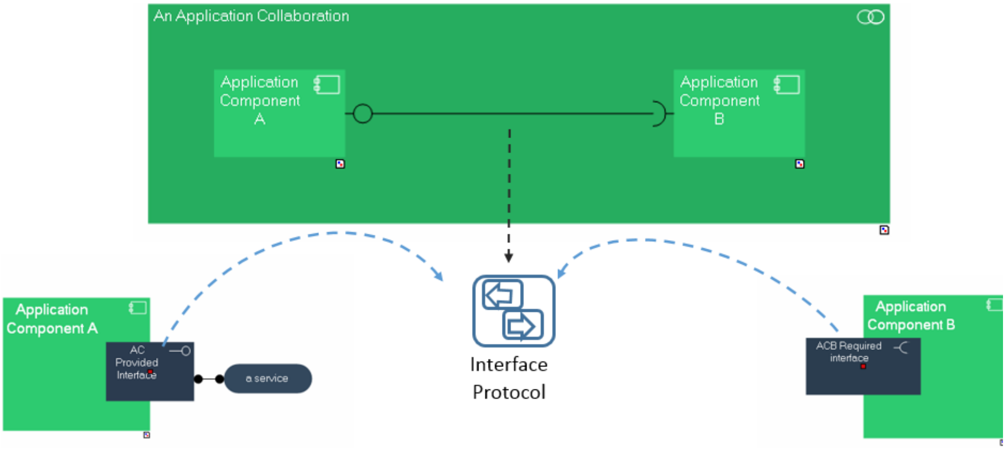
📖 A ***Required Interface*** is a point of exchange by which an agent requests a service from potential suppliers.

☛ *For more details, see "Creating a Provided Interface or Required Interface", page 80.*

The Services usage by components through these **Provided Interfaces** and **Required Interfaces** are captured thanks to **Interactions**.

The **Interaction** uses an **Interface protocol** which is supported by the components **Provided Interfaces** and **Required Interfaces.**

> 📖 *An interaction is defined by the information exchanges by a collaboration of two or more components. It is described by an interface protocol.*
>
> ☛ *For more details, see ."Using Interface Protocols", page 81.*



> ☛ *Interface Procols can be further described by dedicated diagrams; this is available as part of the **HOPEX System Oriented IT Architecture** Product.*

Creating a Provided Interface or Required Interface

The process for creating a **Provided Business Interface** or **Required Business Interface** is identical.

To create a **Provided Business Interface** for a Business Role:

1. Select the Business Role and open its Actor Co-operation viewpoint diagram,
2. In the insert toolbar of the diagram, click the **Provided Business Interface** button.
3. Click in the diagram.
   The **Adding a Provided Business Interface** dialog box appears.
4. Enter the **Name** of the new Interface and click **Add**.
   The **Creation of Provided Interface Protocol** dialog box appears.
5. Click the arrow at the right of the **Interface Protocol** field to define the interface protocol enabling activation of this provided interface.
6. Select **Create Interface Protocol**.
   The creation of interface protocol dialog box appears.
7. Enter the Name of the new Interface Protocol and click **Ok**.
8. Click **OK** to close this dialog box.
   The Interface Protocol is automatically created.
9. Click **OK**.
   The Provided Interface appears in the diagram.

Using Interface Protocols

An Interface Protocol is a model of a contract between entities. This contract is described by exchanges between an initiator role and a contributor role.

Access to the Interface Protocol

An **Interface Protocol** is supported by a **Provided Interface** or **Required Interface** which represents the information exchange channel between the components.

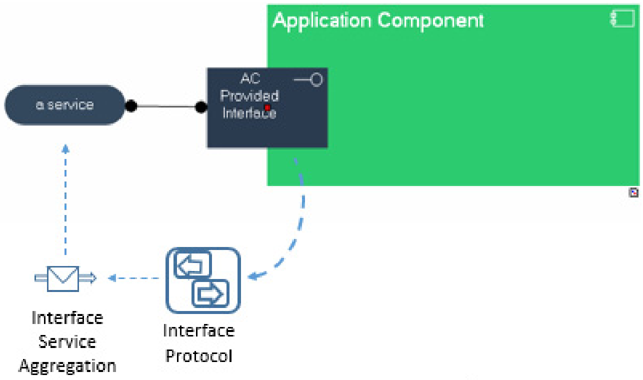
To access to the Interface Protocol properties:

1. Right-click the **Provided Interface** or **Required Interface** which uses the interface protocol and click **Properties**.
2. Click the arrow at the right of the **Interface Protocol** field and click **Properties**.
   The Interface Protocol properties dialog box opens.

   ☛ *The Services of the Interface Protocol are presented in the **Exchange** Tab.*

Describing the Interface Protocol exchanges

The services exposed through **Provided Interfaces** or used through **Required Interfaces** are defined thanks to **Interface protocols** and they are used thanks to interactions.



☛ *An **Interface Protocol** breakdowns into **Services** via **Interface Service Aggregation objects** (automatically created when linking the service to the interface)*

To specify the services exposed through a **Provided Interface** for exemple:

1. Create a service in the diagram.
2. In the insert toolbar of the diagram, click the **Link** button.

3. Click the service and holding down the mouse button, drag the cursor to the Provided Interface.
   The two nodes are highlighted and a dotted line indicates the path that will be taken by the graphic link.

   ☛ *The link can be created from the Interface to the service.*

Creating an Interaction

To create an **Interaction** between two Application Components, for example :

1. In the insert toolbar of the diagram, click the **Link** button.
2. Click the Application Component (service requirer) and holding down the mouse button, drag the cursor to the Application Component to be connected (service provider).
   The two nodes are highlighted and a dotted line indicates the path that will be taken by the graphic link.
3. Release the mouse button.
   The **Creation of Interaction** dialog box appears.
4. Click the arrow at the right of the **Interface Protocol** field to define the interface protocol enabling activation of this interaction.
5. Select the **Interface Protocol** among the possible ones or create a new one.
6. Click **OK**.
   The interaction appears in the diagram.

THE HOPEX METAMODEL FOR ARCHIMATE

This chapter presents the **HOPEX** metamodel used to implement the ArchiMate Framework. The mapping of concepts is illustrated by two metamodels of viewpoint.

CONCEPTS MAPPING

Archimate standard uses itsown terminology. **HOPEX** concepts can be renamed according to the context in which they are used.

> For example, the MetaClass called "Application System" in **HOPEX IT Architecture** standard product is called "Application Collaboration" MetaClass in **Archimate** Terminology.

The renaming mechanism implemented in **HOPEX** enables definition of different names carried by the same concept in its different contexts of use. Each user, depending on his/her profile and the context in which he/she is working, uses terminology with which he/she is familiar. Functionalities proposed here are based on the **Terminology** notion.

> ☛ *For more details, see "Renaming HOPEX concepts" in the technical artical* ***HOPEX Power Studio - Studio****.*

The list of ArchiMate concepts with their **HOPEX** equivalents is presented in the table below.

ArchiMate Concept	HOPEX Concept	Alias	Comment
Business Actor	Org-Unit	Business Actor	
Business Role & Business Collaboration	Business Function	Business Role	The same HOPEX metaclass is used for both concepts, so relations are the same for a role and a collaboration. For example, you can connect a collaboration to an actor.
Business Interface Application Interface Infrastructure Interface	Service Point Request Point	Provided Interface Required Interface	
Business Service Application Service Infrastructure Service	Exchange	Service	The same HOPEX metaclass is used for the three concepts. A specific typing attribute is used to differentiate the instances.
	Exchange Contract	Interface Protocol	Interactions between usage of communicating objects or between interfaces rely on interface protocols; interface protocols are supported by interfaces.

ArchiMate Concept	HOPEX Concept	Alias	Comment
Application Component System Software	Application	Application component	The same HOPEXmetaclass is used for the two concepts. The connexion with an interface is managed by a specific path.
Application Collaboration	Application Collaboration	Application system	The connexion with an interface is managed by a specific path.
Business Process Business Function Business Interaction	Functional Process	Business Process	In HOPEX, this covers the functional aspects of processes.
	Functional Activity	Business Activity	Step in a Business process. Corresponds to a BPMN activity
Business Process	Organizational Process	Org Process	In HOPEX, this defines the implementation of a business process in a given context.
	Operation	Org activity	Step in a Business process. Corresponds to a BPMN activity
Business Event Interface	Event	Business Event	
Business Object	Concept		Note: in process diagrams (business process and org process viewpoints), business objects can be encapsulated within "objects" and linked to the carried products. A specific wizard enables the link creation
Product	Content	Product	Note: contents can be linked to flows for instance to detail the carried information content; products are therefore used in a broader sense of "work product" here
Contract	Not implemented		
Meaning	Not implemented		
Representation	Not implemented		

ArchiMate Concept	HOPEX Concept	Alias	Comment
Application Function Application Interaction Infrastructure function	System Process	IT Function	Note: the alias is kept generic for this can be used for infrastructure as well
	Task	IT Activity	Step in a IT function process; corresponds to a BPMN Activity
Data Object	Entity(DM)	Data Object	In process diagrams (application behavior viewpoint), data objects can be encapsulated within "objects" and linked to the carried products. A specific wizard enables the link creation
	Data Store	Information Store	Information Store can be used within an Application Component to host data objects (via information model)
	Application Data Store	Application Component Information Store	Information Store can be used within an Application Component to host data objects (via information model)
	Data Model	Information Model	An information model is described by the Information Structure viewpoint and contains data objects. It is linked to a Information Store.
Value	Not implemented		
Artifact	Artifact (UML)	Artifact	