ና software AG

Portfolio Management Advanced

Alfabet Reference Manual

Documentation Version Alfabet 10.9.1

CONVENTIONS USED IN THE DOCUMENTATION

Convention	Meaning
Bold	Used for all elements displayed in the Alfabet interface including, for example, menu items, tabs, buttons, dialog boxes, page view names, and commands. Example: Click Finish when setup is completed.
Italics	Used for emphasis, titles of chapters and manuals. this Example: see the *Administration* reference manual.
Initial Capitals	Used for attribute or property values. Example: The object state Active describes...
All Capitals	Keyboard keys Example: CTRL+SHIFT
File > Open	Used for menu actions that are to be performed by the user. Example: To exit an application, select File > Exit
< >	Variable user input Example: Create a new user and enter <User Name>. (Replace < > with variable data.)
	This is a note providing additional information.
	This is a note providing procedural information.
	This is a note providing an example.
	This is a note providing warning information.

TABLE OF CONTENTS

Chapter 1: Introduction to Portfolio Management Advanced

The Portfolio Management Advanced sales package focuses on cost driver analysis, investment prioritization, OPEX optimization and IT forecasting. The capabilities available in the Portfolio Management Advanced sales package allow you to understand the following:

- Which contracts are relevant for the enterprise and which parts of the business are they are relevant for?

- Which contracts are associated with which vendor and provider organization and what risks may be present?

- How are contracts fostering or potentially limiting the consolidation efforts the enterprise is pursuing?

- What cost structures such as cost types and cost centers are impacting the enterprise?

- What are the operational expenditures and lifecycle costs of the IT?

- How much money is spent on a specific business capability or business process?

The capabilities available in the Portfolio Management Advanced sales package provide you with an improved understanding of IT financial performance and optimized IT spending. The Portfolio Management Advanced sales package is used by enterprise architects, program managers, business analysts, and IT controllers:

The purchase of the Portfolio Management Basic package is a prerequisite for the implementation of the Portfolio Management Advanced package.

- Introduction to Portfolio Management Advanced

- Contract and Vendor Management

- OPEX Optimization

- Cost Driver Analysis

Chapter 2:   Contract and Vendor Management

The Contract and Vendor Management capability focuses on the relationships between contracts and the IT architecture. This capability highlights which architecture elements are associated with a specific contract or group of contracts as well as which contracts a specific architecture element is associated with.

The Contract and Vendor Management capability supports the enterprise in managing contracts and the vendors fulfilling the contracts, thus provides transparency to the use of contracts and the enterprise architecture. This capability ensures that the responsibility for contracts or aspects of a contract is understood in the enterprise and that necessary measures required to maintain or terminate contracts are timely and cost-effective. In the context of the contract management functionality, the enterprise can document the deliverables required to fulfill a contractual agreement as well as the architecture elements in the IT landscape that consume the contract deliverables. As a result, contracts can be assessed for their validity in order to maximize the usage of existing contracts and avoid unnecessary costs associated with obsolete or redundant contracts.

A well-established and functional vendor management system is an important part of technology portfolio management, leading to better planning of technology roadmaps for better governance of the enterprise technology portfolio. In the context of the Contract and Vendor Management capability, contracts associated with vendors can be analyzed and understood in order to improve the performance of vendor delivery and pro-actively manage it. Vendor products are managed in terms of their availability and lifecycles in order to provide up-to-date vendor-related information and comprehensive support to the process of managing technology portfolios. By means of the Contract and Vendor Management capability, vendor and contract managers can strike better deals with existing and new vendors, better manage licenses, monitor vendor performance with respect to contracts and negotiate optimal terms and conditions, and support the planned sundowning of a matured or unsuitable technology.

Having transparency between contracts, vendors, and IT architecture enables the business to understand the following:

- Which vendor products are necessary to support the technology road map?

- Which contracts are relevant for a planned change in the architecture (for example, sundowning an application)?

- Which contracts are associated with a specific vendor/provider organization and what risks are implicated?

- How can contracts be changed to mitigate vendor-related risks?

- Where can this information potentially be used to negotiate better discount levels for existing or new contracts?

- Which contracts are relevant for which business process, domain, or organization, etc.?

- When must contracts be extended, renewed or canceled?

- Which vendor products and associated contracts can be consolidated, cancelled, or replaced?

The following information is available regarding the Contract and Vendor Management capability:

- [Methodology: Understanding Contract and Vendor Management](#)

- [Prerequisite: Configuration Requirements for Contract and Vendor Management](#)

- [Understanding Governance and Responsibility for Contracts](#)

- [Creating and Managing Contracts and Contract Items](#)

- [Specifying and Tracking the Contract Deliverables Required to Fulfill the Contract](#)

- [Managing the Costs Associated with Contracts](#)

- [Managing the Contracts in the Enterprise](#)

- [Capturing the Vendors and Vendor Products Relevant to Contracts](#)

- [Working with Vendor Products from the Technopedia® Repository](#)

> ⓘ Please note that a context-sensitive Help is available for each view available in the Contract and Vendor Management capability. You should refer to the Help if you require an explanation about the functionalities and information available in a specific view.

Methodology: Understanding Contract and Vendor Management

A contract is a legal document that stipulates the terms of agreement between organizations or vendors providing products and services and the organizations buying products and services. The following types of contracts are typically relevant to the IT enterprise:

- License, maintenance, and service contracts for applications and technologies

- Operations contracts for devices and associated deployments

- Help desk management contracts for applications and the business processes they enable/support

- Business process outsourcing contracts

- Statements of work contracts managing the execution of projects

- Contractor management contracts on a time and material basis
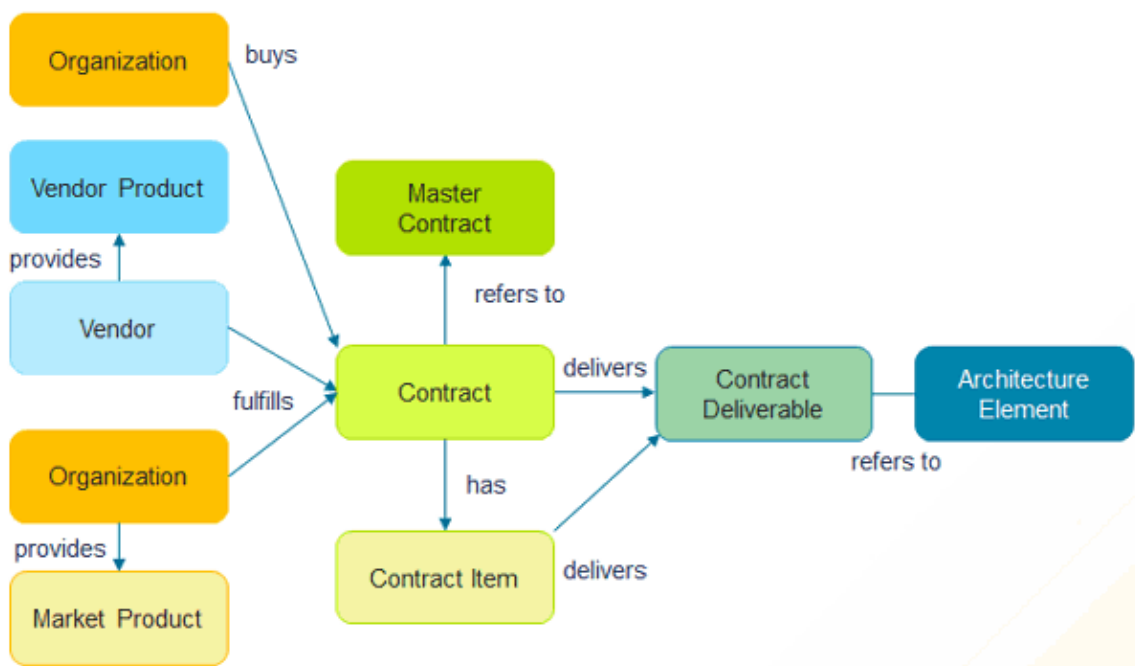
In Alfabet, a contract may be referenced as a master contract by multiple contracts. Each contract may have multiple contract items defined, whereby each contract item is a part of a contract that a specified organization is typically responsible for. For example, a development contract New Trading Infrastructure could have two contract items such as the license purchase of a software product and the maintenance of the software product. Contract items are defined in order to ensure that relevant parts of a contract are maintained or acted upon by the specified organization and to manage the differences in lifecycles for the contract items relevant to the contract. Cost information can be captured for each contract item.

A contract may have contract deliverables that specify the architecture elements and resources to be provided in order to fulfill the contract agreement. For example, the development contract New Trading Infrastructure may have a contract deliverable Trading Software that delivers the application TradeNet.

The Contract and Vendor Management capability allows commercial contracts as well as free and open source contracts to be captured and managed. The contract deliverables for commercial contracts are typically captured directly for the contract or contract item in the *Contract Deliverables Page View* of the relevant contract or contract item. In the case of free and open source contracts, contract deliverables are typically defined from the perspective of a vendor, component, local component or standard platform. In this context, the referenced contract would be considered the free or open source license type being used for the delivery or use of the object. To this end, you would create a contract deliverable in the *Contract Deliverables Page View* of the selected vendor, component, local component, or standard platform that consumes the contract deliverable.

It is recommended that a solution designer configure custom selectors that ensure that the contract stereotypes relevant for free and open software are associated with the contract deliverable.

In general, contract and vendor management is broken down into the following main tasks:

- **Capture and Manage Contracts**: Document contracts and their costs, and associate the costs with the architecture elements that constitute the contract deliverable. Manage the validity and deadline of contracts as well as the lifecycles of their contract deliverables, understand the consumption of contract deliverables such as the number man days in the case of services delivered or the number of licenses in the case of products delivered, manage contract payment schedules, and analyze contracts based on various evaluation criteria.

- **Capture and Manage Vendors**: Govern vendors and vendor products, ensure standardization of vendor product information across the enterprise, analyze contracts with a specific vendor, and understand vendor delivery performance in the context of the contracts.

The implications of contracts can be assessed as part of the project assessment and approval process. Projects can be captured, tracked, and evaluated in the context of the Project Portfolio Governance capability, which is available in the IT Planning Advanced sales package.

Prerequisite: Configuration Requirements for Contract and Vendor Management

In order to work with the Contract and Vendor Management capability, the following configuration may be required:

- If object class stereotypes are to be implemented for the object classes **Contract**, **Contract Item**, **Contract Deliverable**, or **Vendor Product**, these must first be configured in the configuration tool Alfabet Expand.

- If free and open source contracts will be captured, for example, it is recommended that a solution designer configure custom selectors that ensure that the contract stereotypes representing free and open software are associated with the relevant contract deliverable.

- Some fields in editors allow users to select customer-defined values. The enumerations `ContractDeliverableUnit`, `ContractDependencyType`, and `ContractPaymentType` must be configured in the configuration tool Alfabet Expand.

- If vendor products are to be imported via the Technopedia® repository, the XML object **TechnopediaConfig** must be configured in the configuration tool Alfabet Expand.

For details, see the sections *Configuring the Contract Management Capability* and *Configuring Interoperability with Technopedia* in the reference manual *API Integration with Third-Party Components*.

Understanding Governance and Responsibility for Contracts

A number of governance concepts are implemented for contracts:

- **Authorized User**: Each contract, vendor, and vendor product has an authorized user. An authorized user has primary responsibility for the contract/vendor/vendor product and thus has Read/Write access permissions to it. Users may also be assigned to authorized user groups. All users assigned to an authorized user group that has been defined for a contract/vendor/vendor product will have Read/Write access permissions to it. Any contract items defined for a contract will inherit the contract's authorized user definition.

- **Mandates**: Contracts, vendors, and vendor products may be managed in a federated architecture. By means of mandates, it is possible to specify the visibility of individual contracts, vendors, and vendor products in the Alfabet interface for specific users.

- **Object Class Stereotypes**: Object class stereotypes may be configured by your solution designer for the object classes Contract, Contract Item, and Vendor Product. This allows for example for a different governance approach between different kinds of contracts such as Help Desk Contracts and Development Contracts. If object class stereotypes are configured for the object classes Contract, Contract Item, or Vendor Product each stereotype may capture a specified set of attributes, reference data, and class configurations as well as implement a different governance approach.

- **Roles**: A role defines the functional relationship or responsibility that a user or organization has to a contract. In the context of contracts, for example, IT Controller, Application Portfolio Managers, Technology Portfolio Managers, and Contract Managers may be required in order to provide relevant input into the contract in the enterprise. Roles describe responsibilities but they do not authorize access permissions to the contract in Alfabet.

- **Contract Groups**: Contracts may be structured in one or more contract groups. Each contract group has an authorized user and may have authorized user groups. The authorized users of a contract group will have access permissions to all contracts in the contract group.

> (i) Objects in Alfabet are managed by various access permission concepts. For an overview of the access permission and governance concepts in Alfabet, see the section *Understanding Access Permissions in Alfabet* in the reference manual *Getting Started with Alfabet*.

Creating and Managing Contracts and Contract Items

Contracts are captured in the *Contract Schedule Page View* of a contract group in the **Contract Management** functionality.

If object class stereotypes have been configured by your solution designer for the class Contract, you will first be asked to select the stereotype that the contract is based on. The contract is then created and defined by means of the **Contract** editor.

 FIGURE: Contract editor used to create the contract New Trading Infrastructure

The following data is mandatory and must be defined when a contract is created:

• Each contract requires a unique name.

• Each contract requires a buyer. This is the organization that is buying the product that is relevant for the contract.

• Each contract requires a provider. This is the vendor providing a vendor product or an organization providing a market product.

- You should define planned start and end dates specifying the contract's period of validity. The lifecycle of the contract and its contract items and contract deliverables can be viewed in the *Contracts Lifecycle Page View*.

- You may specify a release status, which typically expresses agreement to the state of the documented information. In the example above, a default release status has been configured that is automatically entered when the contract is created. This can be changed in the editor, if needed.

- You may assign a master contract that this contract references.

- You should provide a description of the contract so that other users understand its purpose.

- As the creator of the contract, you are automatically defined as the authorized user per default. The authorized user of the contract can be changed in the **Authorized Access** tab. You can also define any user groups that should have Read/Write access permissions to the contract in the **Authorized Access** tab.

- You may specify the total cost and the monthly cost of the products or services addressed by the contract in the Details tab of the Contract editor, as well as a date when the contract should be reviewed.

    A date monitor can be configured to alert the authorized user when the contract's start date, end date, or review date approaches. For more information about configuring date monitors, see the section *Configuring Monitors to Track Objects in Alfabet* in the reference manual *User and Solution Administration*.

- You may specify the dependencies that the contract may have on other contracts in the *Contract Dependencies Page View*. Documenting contract dependencies ensures that contracts between different contracting parties that have a logical tie are finalized, extended, renewed or cancelled as the need may be.

- You may specify the architecture elements that the contract will deliver. This is described in the section Specifying and Tracking the Contract Deliverables Required to Fulfill the Contract.

Contract items can be defined in order to ensure that relevant parts of a contract are maintained or acted upon by a specified organization and to manage the differences in lifecycles for the contract items comprised by a specific contract. Each contract may have unlimited number of contract items defined. The contract item inherits the authorized user definition from its parent contract. Contract items are defined in the *Contracts Items Page View* of a contract. The following data should be defined when a contract is created:

- Each contract item requires a unique name.

- You should define planned start and end dates specifying the contract item's period of validity. The lifecycle of the contract item can be viewed in the *Contracts Lifecycle Page View* for the parent contract.

- You may specify a release status, which typically expresses agreement to the state of the documented information. In the example above, a default release status has been configured that is automatically entered when the contract is created. This can be changed in the editor, if needed.

- You may specify the organization that is responsible for the contract item.

- You may specify the total cost and the monthly cost of the products or services addressed by the contract item as well as a date when the contract should be reviewed.

- You may specify the architecture elements that the contract item will deliver. This is described in the section Specifying and Tracking the Contract Deliverables Required to Fulfill the Contract.

Specifying and Tracking the Contract Deliverables Required to Fulfill the Contract

In some cases, an architecture element or resource is required in order to fulfill a contract. By defining one or more contract deliverables for a contract or a contract item, you can specify what must be completed or delivered according to the terms of the contract. Any of the following objects in Alfabet may be specified as a contract deliverable:

- application

- business data

- business function

- business object

- business process

- component

- device

- ICT object

- market product

- master platform

- peripheral

- project

- service product

- standard platform

- vendor product

> ⓘ The object classes that you may define as contract deliverables will depend on the sales packages that your enterprise has purchased. For an overview of the object classes available for a sales package, see the chapter *Mapping of Classes to Alfabet Capabilities* in the reference manual *Configuring Alfabet with Alfabet Expand - Appendix*.

In addition to specifying the architecture element, you may need to specify such details as the information about what is consumed by the contract deliverable (for example, the number of man days in the case of services delivered or the number of licenses in the case of products delivered) for the contract or contract item as well as the current status of the delivery and the delivery date. The contract deliverable usage is defined in the *Contract Deliverables Usage Page View* of the architecture element associated with the contract deliverable.

The consumption of the contract deliverables relevant for a contract or contract item can be tracked in the *Deliverable Usage Page View* available for each contract deliverable defined for a contract or contract item.

A date monitor can be configured to alert the authorized user when the delivery date approaches. For more information about configuring date monitors, see the section *Configuring Monitors to Track Objects in Alfabet* in the reference manual *User and Solution Administration*.

Managing the Costs Associated with Contracts

The costs incurred by a contract or contract item can be documented in the Details tab in the editor of the respective contract or contract item. You can capture the total cost and/or monthly cost of the products or services addressed by the contract as well as a date when the contract should be reviewed. A date monitor can be configured to alert the authorized user when the review date approaches.

Helpdesk Contract: Trade*Net Help Desk (new) 2
Contract Payment Schedule

	Name	Volume ▾	Currency	Type	Execution Date	Status
1	2011/07	36,000.00	EUR	Expenditure	24/07/2015	Canceled
2	2011/06	32,000.00	EUR	Expenditure	24/06/2015	Canceled
3	2011/07	25,500.00	EUR	Expenditure	24/07/2015	Executed
4	2011/06	25,000.00	EUR	Expenditure	24/06/2015	Executed
5	2011/08	24,000.00	EUR	Expenditure	24/08/2015	Executed
6	2011/03	24,000.00	EUR	Expenditure	24/03/2015	Executed
7	2011/09	23,600.00	EUR	Expenditure	24/09/2015	Executed
8	2012/02	23,000.00	EUR	Expenditure	24/02/2016	Open
9	2011/11	23,000.00	EUR	Expenditure	24/11/2015	Executed
10	2012/01	22,000.00	EUR	Expenditure	24/01/2016	Executed
11	2011/10	22,000.00	EUR	Expenditure	24/10/2015	Executed
12	2011/02	22,000.00	EUR	Expenditure	25/02/2015	Executed
13	2011/05	21,500.00	EUR	Expenditure	24/05/2015	Executed
14	2011/04	21,000.00	EUR	Expenditure	24/04/2015	Executed
15	2011/01	20,000.00	EUR	Expenditure	25/01/2015	Executed
16	2011/12	19,000.00	EUR	Expenditure	19/12/2015	Executed

FIGURE: Schedule of payments for the contract Trade*Net Help Desk

You can also define a payment schedule for each contract in the *Contract Payment Schedule Page View*. For each payment that you specify, you can capture the status of the payment, the amount and type of payment, as well as the date that the payment should be executed. You can also specify a date monitor to alert about an approaching date that a contract payment should be made. For more information about configuring date monitors, see the section *Configuring Monitors to Track Objects in Alfabet* in the reference manual *User and Solution Administration*.

Managing the Contracts in the Enterprise

Contracts can be stored in a hierarchy of contract groups that allow the contracts to be structured. A contract group may contain multiple contracts. A contract may only be assigned to one contract group. A variety of reports are available that allow you to manage and analyze individual contracts as well as the contracts in a contract group.
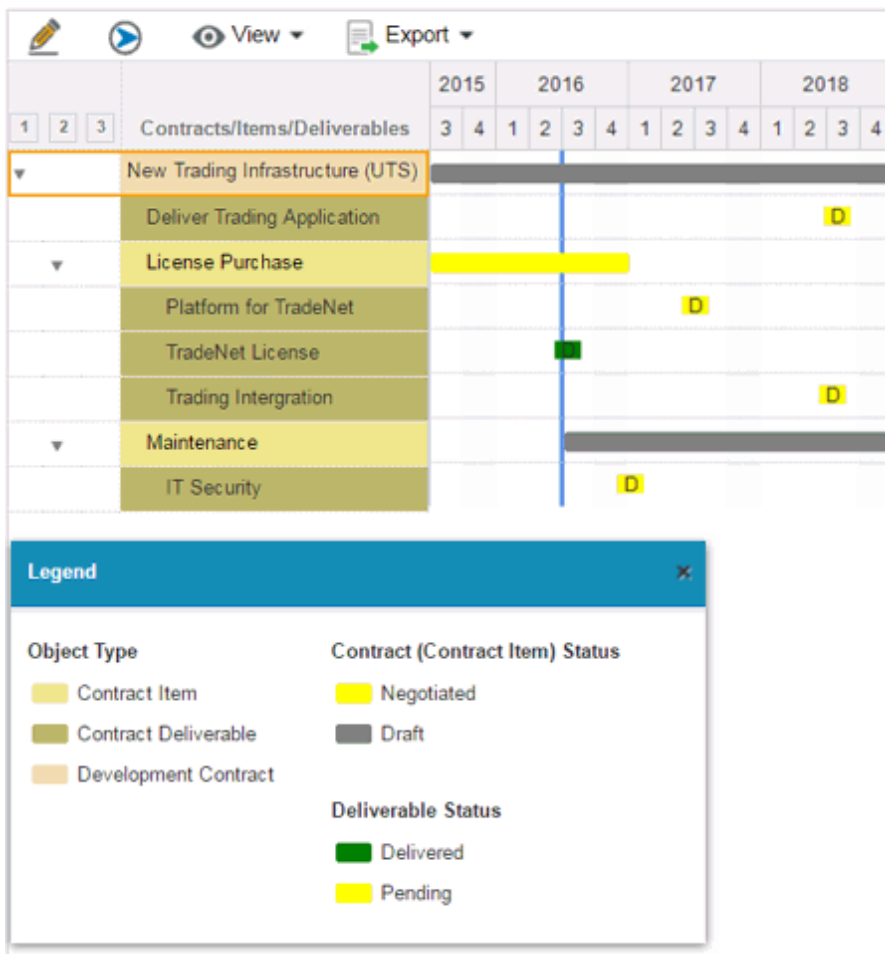
FIGURE: Scheduling and status of contract, contract items, and contract deliverables

- The *Contracts Lifecycle Page View* is available for contract groups as well as individual contracts. These views display the start and end date of the contracts as well as the lifecycle of their contract items and status of their contract deliverables.

- The *Contract Portfolio Page View* provides a configured portfolio that allows the contracts in a contract group to be displayed based on configured criteria. For example, contracts regarding support services could be analyzed based on the number of complaints registered and the average solution time per incident.

- The *Deliverable Usage Page View* provides a report displaying all architecture elements delivered by means of the contract deliverable as well as the information about what is consumed by the contract deliverable (for example, the number of man days in the case of delivered services or the number of licenses in the case of delivered products).

- In addition, your enterprise may configure reports that are specific to the analysis needs of your company. For more information about creating configured reports, see the section *Configuring Reports* in the reference manual *Configuring Alfabet with Alfabet Expand*. For example, a report could be configured to expose which architecture elements are potentially redundantly supported by multiple contracts.

Capturing the Vendors and Vendor Products Relevant to Contracts

Each contract is associated with either a vendor that provides vendor products to realize the contract or an organization that provides market products to realize the contract. The gathering of information about the vendor products in the IT infrastructure is fundamental to contract management as well as technology portfolio governance. The enterprise's vendor products can be captured conventionally or based on products in the Technopedia® repository.

To capture organizations, you must have access to the Portfolio Management Basic sales package. To capture market products, you must have access to the Business Model Definition capability, which is part of the IT Planning Basic sales package.

The vendor can supply multiple vendor products, which are the goods or services that are provided to the company. Each vendor product may be associated with one or more technical components. For example, the vendor product Oracle RDBMS may be split into the components Oracle 11i Server, Oracle 11i Client, and Oracle 11i OCL. Further, a component may reference a vendor product, thus indicating that it has been derived from the vendor product. Each vendor product may be assigned to one vendor product category, which bundles and classifies content-specific vendor products.

You can manually capture vendors and the vendor products that they provide in the **Vendors** explorer or via the import of vendor products from the Technopedia® repository. When you manually capture a vendor, you must specify the name of the vendor along with optional information such as contact information, the vendor's Website, and a description of the vendor. If object class stereotypes have been configured by your solution designer for the class Vendor, you will first be asked to select the stereotype that the vendor is based on. As the creator of the vendor, you are automatically defined as the authorized user per default.

Vendor products can be manually captured in the *Vendor Products Page View* of a vendor. If object class stereotypes have been configured by your solution designer for the class Vendor Product, you will first be asked to select the stereotype that the vendor product is based on. The vendor product is then created and defined by means of the **Vendor Product** editor.

FIGURE: Vendor Product editor used to create the vendor product Oracle Database

The following data is mandatory and must be defined when a vendor product is created:

- Each vendor product requires a unique name and version.

- You should define planned start and end dates specifying the vendor product's period of validity. The lifecycle of the vendor product can be viewed in its *Lifecycle Page View*.

- You may specify a release status, which typically expresses agreement to the state of the documented information. In the example above, a default release status has been configured that is automatically entered when the vendor product is created. This can be changed in the editor, if needed.

- You may assign the ICT object that owns the vendor product.

- You should assign the vendor that is providing the vendor product.

- You may assign a domain that is the primary business domain that the vendor product is assigned to.

- You should provide a description of the vendor product so that other users understand its purpose.

- As the creator of the vendor product, you are automatically defined as the authorized user per default. The authorized user of the vendor product can be changed in the **Authorized Access** tab. You can also define any user groups that should have Read/Write access permissions to the vendor product in the **Authorized Access** tab.

Working with Vendor Products from the Technopedia® Repository

Software AG provides an interface to the Technopedia® repository of software and hardware products. The Technopedia capability allows vendor products to be captured in a standardized manner and the technology information to be aligned across the enterprise. The catalog of structured information about the IT infrastructure ensures a unified language and discipline to manage and plan the technology portfolio. With consistent naming and standardized data, the complexity of the IT landscape can be reduced, overlapping technologies eliminated, and existing and known technologies reused. If interoperability with Technopedia is supported by your enterprise, Alfabet users will be able to create vendor products in Alfabet based on Technopedia software products and/or hardware products.

Technopedia® has a two-level taxonomy comprised of product categories and sub-categories that contain the Technopedia products. Software AG does not own the Technopedia® taxonomy and using the Technopedia® taxonomy may result in a structure that is different from your enterprise's technology domain structure. Typically, the entire repository of Technopedia® product categories would be imported to Alfabet. Existing vendor product categories already defined in Alfabet will remain in the Alfabet database but will not be displayed in the Technopedia selector available in the Alfabet user interface that allows users to import Technopedia hardware and software products. The top level of the Technopedia® product categories will be imported to the top level of the vendor product hierarchy and subordinate categories will be displayed below their parent vendor product category.

The imported vendor product categories will initially have no vendor products assigned to them. In other words, the vendor products must be explicitly created based on the software/hardware product in the Technopedia repository. The software and hardware products in the Technopedia® repository can be selected on a category-by-category basis in order to create vendor products in Alfabet. The vendor products that are created based on Technopedia products are saved to the Alfabet database and can be further defined as needed. When the new vendor product is created, the vendor defined for that Technopedia product will be automatically assigned to the new vendor product in Alfabet. If the vendor does not already exist in the Alfabet database, it will be created in Alfabet along with the new vendor product. If an existing vendor product is already mapped to a Technopedia software or hardware product and the user selects the same

Technopedia product and its `id` property matches the `TP_ID` property of the existing vendor product, then the existing vendor product will be updated and a new vendor product will not be created.

> This documentation primarily describes the creation of vendor products based on Technopedia software and hardware products. However, some enterprises may prefer to implement the Technopedia capability to create components rather than vendor products. This will depend on the methodology implemented in your enterprise. Please note the following regarding using components instead of vendor products:
>
> - The Technopedia® product categories will be mapped to component categories and the hardware and software products will be mapped to components.
>
> - The object class `Component` must be specified in the XML attribute `ClassMapping` in the XML object **TechnopediaConfig**.
>
> - For each component created in Alfabet based on a Technopedia software of hardware product, an object in the object class `VendorProduct` will first be created and then an object in the object class `Component` will be created. For more information, see the section *Understanding the Mapping of Technopedia Products to Vendor Products in Alfabet.*
>
> - The relevant page views must be made available to the user community as described in the section *Making the Technopedia Capability Available to the User Community*.
>
> - The predefined ADIF import scheme `ALFABET_TECHNOPEDIA_UPDATE` is only relevant for the update of Alfabet vendor products and ICT objects. You must configure an ADIF import scheme in order to synchronize Alfabet components with the current information in Technopedia.

A subscription concept can be implemented for ICT objects that are based on Technopedia products in order to simplify the maintenance of the enterprise's technology catalog. If the ICT object is subscribed to the Technopedia product that it is based on, vendor products can be automatically created and maintained for all relevant release versions of the Technopedia product that the ICT object is based on. To implement the subscription concept for an ICT object, the **Is Subscribed** checkbox must be selected in the **Technopedia** tab of the **ICT Object** editor. The **Is Subscribed** checkbox specifies that the ICT object has a subscription to the Technopedia product that it is based on. The release version level (**All Releases**, **Major Releases**, **Minor Releases**) of the Technopedia product that shall be used to create the vendor products must be specified in the **Subscription Level** field. The ADIF job `ALFABET_TECHNOPEDIA_UPDATE` must be executed by a user with an administrative user profile in order to create the vendor products. Vendor products will be created for all relevant release versions of the Technopedia product that the ICT object is based on. The vendor products will be added to the **Vendor Products** page views for the ICT object they are associated with.

Vendors can also be created in Alfabet based on Technopedia manufacturers.

> The configuration of the Alfabet solution to interface with the Technopedia API as well as a subscription to Technopedia services is required in order to access the Technopedia repository and create vendor products based on the software products and hardware products stored in Technopedia.

The following workflow is recommended when working with Technopedia:

- **Configuring Technopedia Interoperability** : Configuration is primarily carried out in the XML object **TechnopediaConfig**. If your enterprise plans to base new vendor products on both hardware products and software products from Technopedia, then it is recommended that object class stereotypes are configured for the object class `VendorProduct`. For more information about the

configuration requirements as well as the mapping of Technopedia classes and attributes to Alfabet classes and attributes, see the section *Configuring Interoperability with Technopedia* in the reference manual *API Integration with Third-Party Components*.

- **Importing Vendor Product Categories**: Typically, the entire repository of Technopedia® product categories should be imported to Alfabet. This is carried out in the *Root Categories Page View* page view available on the root node of the **Vendor Products** explorer. Each vendor product category will be associated with a Technopedia ID. While it is technically possible to create vendor product categories in Alfabet and create new vendor product categories by importing product categories from Technopedia®, it is recommended that your enterprise use only one source of vendor product categories.

  ⓘ Please note that the product category hierarchy in Technopedia® consists of two levels and that software products and hardware products are assigned only to the leaf-level product categories. The top level of the Technopedia® product categories will be imported to the top level of the **Vendor Products** explorer in the Alfabet interface and the subordinate product category will be nested below. Vendor product categories that have been imported via Technopedia® will initially have no vendor products assigned to them. In other words, you must explicitly create the vendor products in Alfabet based on the Technopedia products for each relevant vendor product category, as needed. Users may assign new vendor products based on Technopedia products to any vendor product category in Alfabet.

- **Creating Vendor Products Based on Technopedia Software Products**: New vendor products based on Technopedia software and hardware products are created in the *Vendor Products Page View* available for a vendor product category as well as for an ICT object, domain, component, or vendor. The vendor product and vendor can be further specified as needed in the context of the Alfabet user interface. For more information about the mapping of properties, see the section *Configuring Interoperability with Technopedia* in the reference manual *API Integration with Third-Party Components*.

- **Synchronizing Vendor Products in Alfabet with Technopedia Products** : A predefined ADIF import scheme `ALFABET_TECHNOPEDIA_UPDATE` is available to updated the vendor products and vendors based on Technopedia products with the current data in the Technopedia repository. When executed, the ADIF import scheme reads the configuration specified in the XML object **TechnopediaConfig** and updates any vendor products and vendors in Alfabet that have been changed since the last time the ADIF import scheme `ALFABET_TECHNOPEDIA_UPDATE` was executed. For more information about updating the vendor products and vendors based on Technopedia products as well as the criteria that must be met for the update, see the section *Updating Technopedia Products in Alfabet via the ADIF Import Scheme ALFABET_TECHNOPEDIA_UPDATE* in the reference manual *Configuring Alfabet with Alfabet Expand*. Please note that a license to the tool Alfabet Data Integration Framework (ADIF) is necessary to synchronize the vendor products with Technopedia products. The Technopedia products can be synchronized by a user with an administrative user profile in the *ADIF Jobs Administration Functionality*. For more information, see the section *Executing and Controlling ADIF Jobs* in the reference manual *User and Solution Administration*.

Chapter 3: OPEX Optimization

Mapping available financial data to the applications in the IT landscape provides insight into which applications are driving costs. Abstraction and aggregation of this financial data helps the enterprise understand the costs required to support its business processes and business capability, consequentially provides the intelligence required for making financially sound business decisions on IT and optimizing the money spent on IT operations.

In order to understand and control operational expenses (OPEX), cost information and the relationships between IT and business processes must be understood by enterprise architects, application owners, IT controllers, and CFOs. Understanding the costs incurred to fulfill the enterprise's business processes and business capabilities is the starting point for application consolidation and the reduction of further costs. By capturing and understanding the costs of the application architecture, it is possible to understand the true cost of IT support to business processes or business capabilities, identify redundant costs, reduce operating expenses, and improve efficiency.

OPEX optimization activity supports the analysis of service level agreements in order to optimally align costs with business priorities.

To capture service level agreements (SLAs), you must have access to the Service Product Portfolio Governance capability, which is part of the Portfolio Management Complete sales package.

The following information is available regarding the OPEX Optimization capability:

* Methodology: Understanding OPEX Optimization

* Prerequisite: Configuration Requirements for Cost Management

* Managing Costs via Cost Centers

* Analyzing ICT Object Budgets

* Analyzing Application Budgets

* Analyzing Service Contract Costs for Organizations

Please note that a context-sensitive Help is available for each view available in the OPEX Optimization capability. You should refer to the Help if you require an explanation about the functionalities and information available in a specific view.

Methodology: Understanding OPEX Optimization

In general, OPEX optimization is broken down into the following main tasks:

* **Cost Structure Definition**: The definition of cost types and configuration of cost management functionalities is foundational to capturing and analyzing OPEX costs. Operational cost types must be configured and can consist of a subset of cost types such as Hardware Replacement Costs, Maintenance Costs, Software Subscriptions, Other Deployment Costs, and Other Operational Costs. If you plan to capture your OPEX costs via cost centers, then cost center types and cost centers must be configured.

* **Document and Analyze OPEX Costs**: Depending on the company's level of maturity, Alfabet provides various means to capture and analyze the operating costs of the business' IT. For

organizations with a low-level of transparency in the IT landscape, this could be executed by defining cost center budgets and distributing the cost center budget across the architecture objects that it owns. For organizations with an increasing level of maturity, this could mean capturing and managing operational costs on the level of either ICT objects or applications and their deployments.

- **Plan Application Budgets**: Once accrued costs and current application budgets are documented and understood, the enterprise can plan future application lifecycle costs as well as develop a sound basis for application consolidation and scenario planning. An understanding of expected application landscape costs lays the groundwork of an informed approach to the management of and optimization of capital expenditure (CAPEX).

Alfabet provides various means to understand the operating costs of the business' IT. The method that you choose for your enterprise will usually depend on the enterprise's level of maturity. Preliminary decisions should be made about the granularity of the information that is available for the IT landscape. The most significant question to consider is whether operational costs can be gathered on the application level or on the ICT object level? For example, are all significant application versions and their deployments captured in the IT inventory? If this is not the case, perhaps your enterprise is able to formulate the budgetary needs of the application landscape via ICT objects. This method requires less detailed information about individual application versions and allows costs to be captured in a more general manner. In this case, you must consider which architecture artifacts the ICT objects shall be defined for (for example, should the ICT objects bundle applications, standard platforms, components, vendor products)? Alternatively, if your enterprise has a very low level of maturity in terms of understanding the IT landscape, you can assess OPEX costs by means of cost centers, which allow the operational costs of individual architecture objects to be generally estimated as a rough percentage of the cost center budget.

> OPEX costs are captured on a yearly basis only. Please note that costs can only be captured on a monthly basis in the context of projects via the *Cash Out Planning Page View*. The capabilities focused on project costs, project planning, scenario management, and CAPEX optimization activities are addressed in the context of the IT Planning Advanced and IT Planning Complete sales packages.

Low-level maturity: In an organization with a low level of maturity, the IT architecture is typically not thoroughly documented and therefore costs cannot be captured consistently for individual architecture objects. In this case, costs can be managed via cost centers which allow you to centrally define the costs related to a set of architecture objects such as applications, ICT objects, or deployments for a specified period of time.
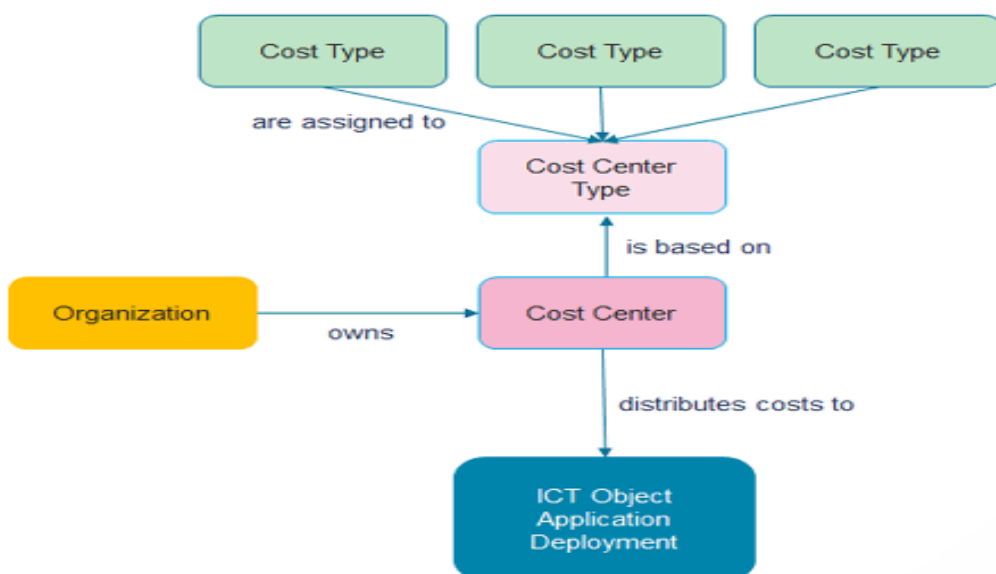
Each cost center is based on a cost center type, which serves as a template of cost types. Applications, ICT objects, and deployments can then be assigned to the cost center.

For example, several cost centers for application-related costs such as Trading Infrastructure, Trading Floor Transformation Costs, and Service and Support Costs are created. The cost center Trading Infrastructure and Trading Floor Transformation Costs are based on the cost center type Operations. These cost centers will have a standardized set of cost types available that are relevant for application-related operations. Each cost center may have additional cost types explicitly added that are not included in the cost center type Operations in order to capture, for example, development-specific costs (for example, Third-Party Licenses) or maintenance-specific costs (On-Site Consultation Fees).
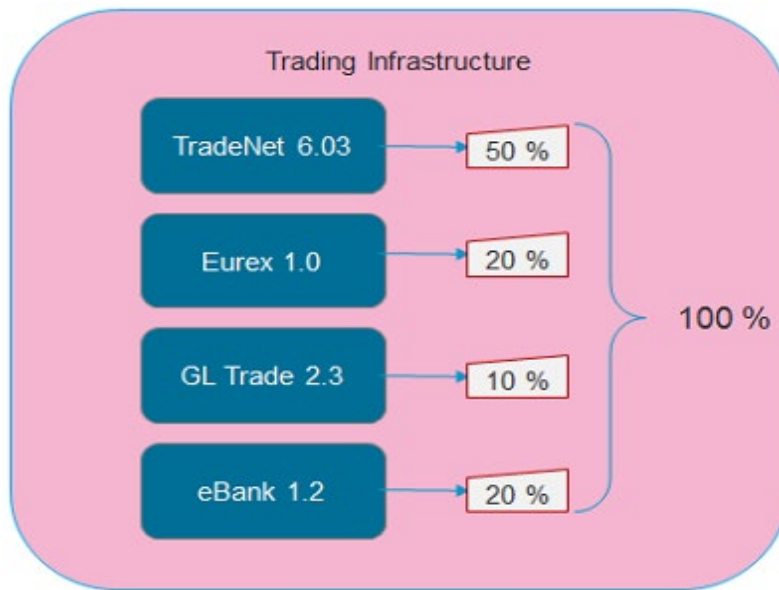

FIGURE: Manual allocation of application costs assigned to the cost center Trading Infrastructure
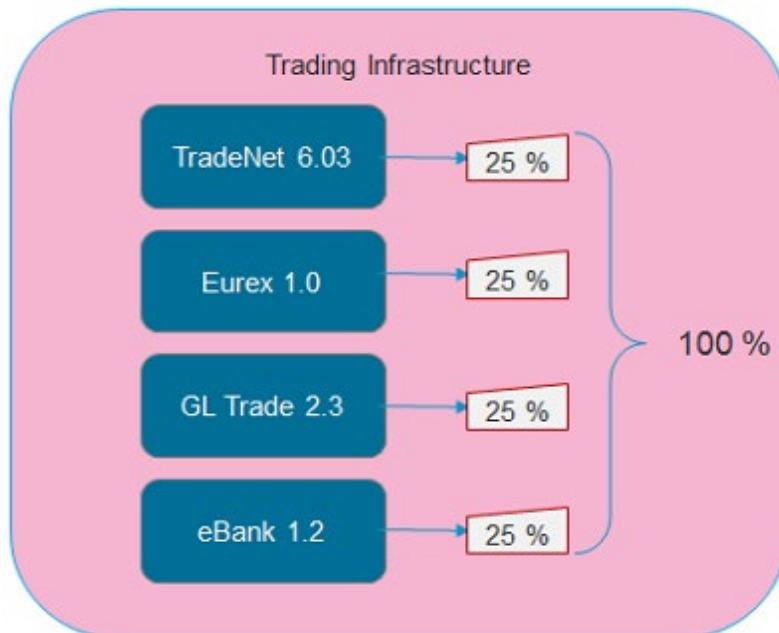

FIGURE: Automatic allocation of application costs assigned to the cost center Trading Infrastructure

The costs of each architecture element can be manually specified as a percentage of the costs center's budget or the percentage of IT costs can be equally distributed across all architecture objects assigned to

the cost center. The specified percentage of the cost center's budget is then calculated and distributed to the *Operational Expenses Page View* of the relevant architecture object, where it can be further refined as needed.

A cost center can also be used to define costs for service products and projects. To capture service products and manage their costs, you must have access to the Service Product Portfolio Governance capability, which is part of the Portfolio Management Complete sales package. To capture projects and plan their budgets, you must have access to the Project Portfolio Governance capability, which is part of the IT Planning Advanced sales package.

Medium-level maturity: In an organization with a medium level of maturity, the IT architecture is documented to a certain degree, although all application versions may not be necessarily known or captured in the IT inventory. In this case, costs can be defined for ICT objects that may or may not be associated with actual application versions. If costs are assigned to ICT objects, then costs should not be assigned to applications.
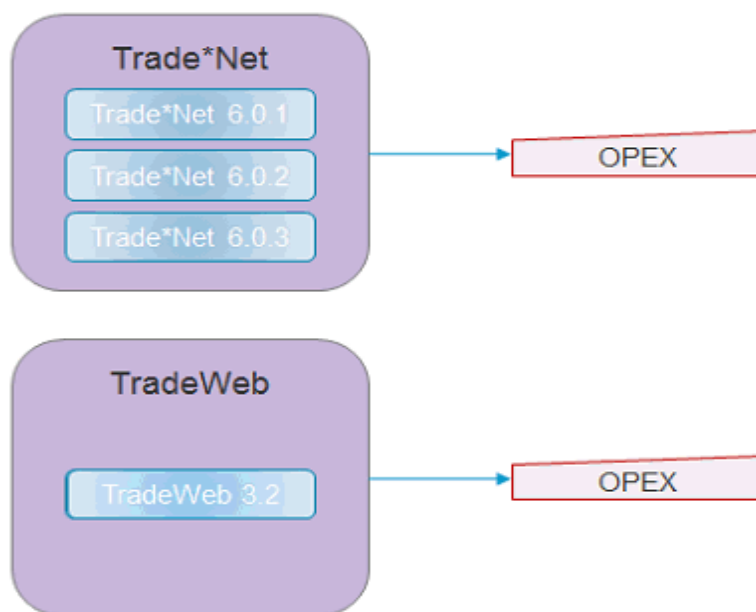

FIGURE: OPEX costs defined explicitly for the ICT objects Trade*Net and TradeWeb

Please note that any costs assigned to an application will be aggregated to the ICT object that the application is associated with. Therefore, as a rule, costs should not be assigned to both applications and ICT objects.

High-level maturity: In an organization with a high level of maturity, most applications are known and captured in the IT inventory. In this case, you can assess the lifecycle costs of application versions, document the percentage of each application's costs that are incurred by the business services it provides, and plan application budgets.

In some cases, the organization may even have specified the deployments of their applications. In this case, costs can be defined for application versions as well as their deployments.
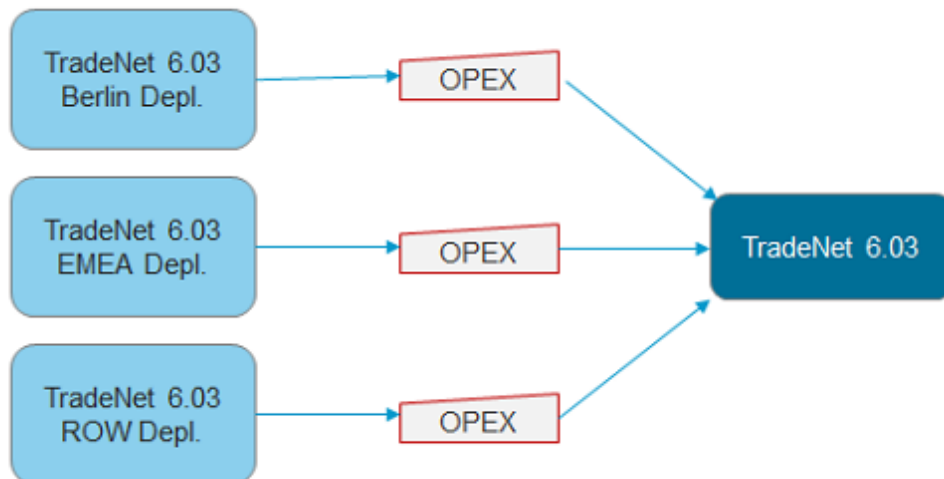
FIGURE: OPEX costs aggregated from application deployments to the application TradeNet

If costs are also assigned to deployments, the deployment costs will be aggregated to the associated application. The cost types captured to maintain and operate applications and deployments may be different. For example:

- Application-related costs such as Development, Maintenance, Help Desk, License Fees, etc. should be captured on the application.

- Deployment-related costs such as Operations, Hardware, Energy, Data Center Fees, etc. should be captured on the deployment.

> Please note that all deployment and application costs will be aggregated to the associated ICT object. Therefore, as a rule, costs should not be assigned to both applications/deployments and ICT objects.

Prerequisite: Configuration Requirements for Cost Management

The following is required in order to work with the cost management capabilities:

- The fiscal year for the cost centers in your enterprise must be configured by your solution designer in the XML object **CostManagerDef** in the configuration tool Alfabet Expand. For more information, see the section *Configuring the Fiscal Year for Cost Reporting in Your Enterprise* in the reference manual *Configuring Alfabet with Alfabet Expand*.

- The currency and currency units displayed on all page views and editors in which costs are captured and visualized is configured in the **Reference Data** functionality. For more information about the configuration of currencies, see the chapter *Configuring Currencies and Currency Exchange Rates for Cost Management Capabilities* in the reference manual *Configuring Evaluation and Reference Data in Alfabet*.

- Cost types that are relevant to the cost centers in your enterprise must be configured in the **Reference Data** functionality. The cost types must be assigned as **Maintenance Cost Types** to the class **Project** and not (**Project** (< **ProjectStereotype** >) in the **Class Configuration** functionality. All cost types assigned to the class **Project** will be available for definition in the *Operational Expenses Page View* for the applications, ICT objects, deployments, service products, and projects.

The creation of cost types is explained in the section *Specifying Cost Types for Cost Management* in the chapter *Configuring Cost Types and Income Types for Cost Management Capabilities* in the reference manual *Configuring Evaluation and Reference Data in Alfabet*.

- The relevant cost definition categories (`Request` (planned costs), `Budget` (endorsed, approved costs), `Current` (accrued costs), `Obligation` (costs of ordered but yet unpaid services), and `Forecast` (expected) must be specified in the XML element `ArchitectureCosts` in the XML object **CustomUnitDef** in the configuration tool Alfabet Expand. The relevant cost definition categories will be available in the *Operational Expenses Page View* for applications, ICT objects, deployments, service products, and projects *Configuring the Editability of Costs for Architecture Objects* in the reference manual *Configuring Alfabet with Alfabet Expand*.

If you plan to capture OPEX costs by means of cost centers, the following configuration is also required:

- Cost center types that are relevant to the cost centers in your enterprise must be configured in the **Reference Data** functionality. The cost types created in the previous step must be assigned to the cost center types. This is described in the section *Configuring Cost Center Types* in the reference manual *Configuring Evaluation and Reference Data in Alfabet*.

- The relevant cost definition categories (`Request` (planned costs), `Budget` (endorsed, approved costs), `Current` (accrued costs), `Obligation` (costs of ordered but yet unpaid services), and `Forecast` (expected) must be specified in the XML element `CostCentreCosts` in the XML object **CustomUnitDef** in the configuration tool Alfabet Expand. For more information, see the section *Configuring the Editability of Costs in Cost Centers* and *Configuring the Editability of Costs for Architecture Objects* in the reference manual *Configuring Alfabet with Alfabet Expand*.

Managing Costs via Cost Centers

A cost center is a means to centrally define costs for a specified period of time and allocate them to a group of architecture objects that the cost center is responsible for. Cost centers are created in the **Cost Centers** functionality. Each cost center will be based on a configured cost center type that bundles cost types. This is described in detail in the section *Creating Cost Centers and Specifying Cost Allocation* in the reference manual *Configuring Evaluation and Reference Data in Alfabet*.

> The tool ADIF provides a batch mechanism to update object assignment to cost centers as well as to update cost allocation to architecture objects. To do this, you must have access to the ADIF functionality. For more information, see the section *Predefined ADIF Schemes* in the reference manual *Alfabet Data Integration Framework*.

The following steps are recommended to manage costs via cost centers in your enterprise:

1) **Create costs centers**: This is carried out in the **Cost Management** functionality. When you create the cost center, you must first select the configured cost center type that it should be based on. The following data should be defined for each cost center:

- Each cost center requires a unique name.

- Each cost center requires planned start and end dates because the allocation of architecture objects to cost centers can change over time. It is therefore common practice to create a new set of cost centers for each fiscal year. Cost centers created for a specified fiscal year can be bundled in cost center groups.

- Each cost center requires an organization to be specified that owns the cost center.

- Should the cost center costs be equally distributed to all architecture objects assigned to the cost center or should the costs of the architecture objects be manually defined for the cost center? If the **Equal Allocation** checkbox is checked, the costs will be equally distributed (as a percentage) across all objects assigned to the cost center. If it is not checked, the **Object Allocation** editor will be available in the *Objects Page View* of a cost center and you can manually adjust the percentage of cost distribution to each individual object.

- You should provide a description of the cost center so that other users understand the purpose of the cost center.

- If you plan to interface with an ERP system:

  - You may specify the ID of the cost center in the ERP system. This will allow you to communicate and map costs in the ERP system.

  - You may specify the ID of the instance of the ERP system that manages the cost center. This will allow you to capture multi-instances or multi-mandate ERP installations in your corporation.

2) **Specify the cost center's budget**: In the *Cost Accrual Page View*, define the costs for the relevant cost types for the selected fiscal year.

> The cost center budget for 2016 and 2017 defined in the **Cost Center Costs** editor available in the *Cost Accrual Page View*.

3) **Specify the architecture objects owned by the cost center**: Once the cost center budget is created, you should assign the relevant architecture objects to the cost center in the *Objects Page View*. The authorized user of the cost center may add the ICT objects, applications, and deployments to the cost center. The authorized user of the cost center does not necessarily need to have Read/Write access permissions to the architecture objects being added to the cost center.

4) **Allocate the costs to the architecture objects**: Return to the *Objects Page View*. If you specified that the costs should be equally distributed among all architecture objects that the cost center owns, then you must manually adjust the percentage values for the relevant architecture objects. The sum of all cost proportions for a cost center must equal 100.

> The cost allocation of the three applications assigned to the cost center is manually defined in the **Object Allocation** editor available in the *Objects Page View*.

5) **Refine the operational costs of the architecture objects**: If necessary, you can open the *Operational Expenses Page View* for each application, ICT object, and deployment to view the budget values distributed to each architecture object based on the allocation defined. The operational expenses may be refined for each architecture object, as needed.

> The cost allocation defined for the applications in the context of the cost center is passed on to each application's budget. The percentage allocation of the cost center budget appears as a monetary value for the cost types Internal Maintenance Costs and Software Subscriptions, which were defined for the cost center. This can be edited in the **Maintenance Costs** editor available in the *Operational Expenses Page View* of the relevant application.

Maintenance Costs				
	Currency values in T$		Years	
			2016	
	Cost Type	Budget		Curren
1 ▼	Operational Expenditure			
2	Hardware Replacement Costs			
3 ▼	Maintenance Costs			
4	Internal Maintenance Costs	4860		4116
5	External Maintenance Costs	920		927
6	Other Deployment Costs			
7	Other Operational Costs	390		395
8	Software Subscriptions	2120		1914

Analyzing ICT Object Budgets

In an organization with a medium level of maturity, the IT architecture may be partially documented but not enough information is known about the application versions to accurately and consistently understand, capture, and plan the application architecture budget. If this is not the case, perhaps your enterprise is able to formulate the budgetary needs of the application landscape via ICT objects. This method requires less detailed information about individual application versions and allows costs to be captured in a more general manner.

ICT objects represent a controlling and planning view of the IT architecture. An ICT object (ICT = In-for-mation and Communication Technology) is an abstract object that represents applications regardless of their versioning and is a means to plan and control costs related to the application and its infrastructure. The use of ICT objects is advantageous in that the planner must not initially commit him/herself to a certain version of the application. By means of the ICT object, portfolio managers can understand the operating costs of the application. Later, at the stage of detailed planning, the ICT object can be replaced by a spe-cific concrete version. The ICT object is owned by an organization that is usually responsible for the budget of the architecture elements assigned to the ICT object.

In this case, costs can be defined for ICT objects, which can represent one or more applications regardless of their versioning as well as other architecture elements that may be related for business or financial rea-sons. In this case, you must consider which architecture artifacts the ICT objects shall be defined for. For example, should the ICT objects bundle only applications or also standard platforms, components, or ven-dor products that constitute the application's technological infrastructure? The use of ICT objects for cost management is advantageous in that the application architect or planner must not initially commit him/herself to a certain version of an application, for example. Later, at the stage of detailed planning, the ICT object can be replaced by a concrete application version. The ICT object is owned by an organization that is usually responsible for the budget of the architecture elements assigned to the ICT object.

If your enterprise chooses to capture and analyze architecture costs by means of ICT objects, then costs should only be captured on the level of ICT objects. If costs are also defined for applications or their deploy-ments, these costs will be automatically aggregated to the ICT object owning the applications, thus

skewing the ICT object budget. Therefore, as a rule, costs should not be assigned to both ICT objects and applications/deployments.

The following steps are recommended to manage costs via ICT objects in your enterprise:

1) **Capture ICT objects**: The creation of ICT objects and the basic information that should be captured from them is described in detail in the section *Capturing ICT Objects to Understand IT Costs* in the reference manual *Enterprise Architecture Management*.

2) **Specify the architecture objects owned by the ICT object**: You may assign the relevant applications, standard platforms, components, and vendor products in the relevant page views in the object profile of the ICT object. The authorized user of the ICT object does not necessarily need to have Read/Write access permissions to the architecture objects being added tot he ICT object. Please note that if costs are captured for applications, they will be aggregated to the ICT object. Costs cannot be captured for standard platforms, components, and vendor products.

> (i) If costs are aggregated from applications to the ICT object, this can be analyzed in the *Operating Expenses Page View*, which provides information about the accrued operating costs for a selected year in a the form of a bar chart.

3) **Specify the ICT object's budget**: In the *Operational Expenses Page View*, define the accrued operating costs for the previous year and the planned costs for the current year for the relevant cost types for the ICT object.

4) **Analyze the IT budget based on ICT object costs**: Various analytical views are available to understand and review IT costs. Analysis may be carried out for a specific ICT object or for a set of ICT objects. ICT objects can be assigned to ICT object groups, which logically structures ICT objects. There may be various ways to logically structure ICT objects so an ICT object may be assigned to multiple ICT object groups. ICT objects may also be assigned to ICT object categories, which classify content-specific ICT objects in a hierarchy, if necessary. Each ICT object may be associated with only one ICT object category:

   - The *Lifecycle Costs Page View* and *Lifecycle Costs Chart Page View* display the costs of the ICT object over the entire lifecycle (start date to end date) of the ICT object. Please note that if costs have been defined for applications, they will be aggregated from applications to the ICT object and included in the cost amounts displayed here.

   - The *Operating Expenses Page View* provides information about the accrued operating costs for the previous year and the planned costs for the current one (for the selected category and its subcategories <3>) This report can be used to analyze the costs of IT support for business domains.

   - The *Cost Aggregation Page View* provides information about the aggregated costs of all ICT objects in an ICT object category or ICT object group over the selected period.

   - In addition to the standard reports listed above, your enterprise may have configured reports that are specifically geared to the assessment and reporting needs of your enterprise. Your enterprise must purchase the relevant license to the expansion set Custom Reports. For more information about configuring reports, see the section *Configuring Reports* in the reference manual *Configuring Alfabet with Alfabet Expand*.

Analyzing Application Budgets

In an organization with a high level of maturity, the application architecture is typically well understood and the most significant applications in the IT inventory are documented. If this is the case in your enterprise, you can capture and plan the IT budget based on the lifecycle costs of the applications currently in operation as well as their planned versions. In addition, you can document the percentage of each application's costs that are incurred by the business services it provides.

Your enterprise may also define the technical deployment of each application. In this case, you can also document the costs of the application deployments. Please note that all deployment and application costs will be aggregated to the associated ICT object. Therefore, as a rule, costs should not be assigned to both applications/deployments and ICT objects. The costs defined for the deployments will be aggregated to the application that they have been defined for. The cost types captured to maintain and operate applications and deployments may be different. For example:

- Application-related costs such as Development, Maintenance, Help Desk, License Fees, etc. should be captured on the application.

- Deployment-related costs such as Operations, Hardware, Energy, Data Center Fees, etc. should be captured on the deployment.

The following steps are recommended to manage costs via applications in your enterprise:

1) **Capture applications**: The creation of applications and the basic information that should be captured for them is described in detail in the section *Application Architecture Definition* in the *Enterprise Architecture Management* reference manual.

> You can also define the technological deployments for each application and capture the costs for the deployments. The specification of application deployments is described in the section *Defining the Operational Deployment of an Application, Component, or Standard Platform* in the *Enterprise Architecture Management* reference manual.

2) **Specify the application's budget**: In the *Operational Expenses Page View*, define the accrued operating costs for the previous year and the planned costs for the current year for the relevant cost types for the application.

3) **Specify the costs of the application deployments**: If you are analyzing the costs of applications including their deployments, you must capture the deployment costs in the *Operational Expenses Page View* of each deployment. The deployment's costs can be aggregated to the application that the deployment is defined for. To view a report displaying the costs of all deployments assigned to a selected application, see the *Consolidated Operational Expenses Page View* in the object profile of the relevant application.

4) **Specify the allocation of the application budget to the business services it provides**: In the *Business Services Page View*, you can define or edit the percentage of the applications costs that are covered by the business services.

> The **Business Service Cost Allocation** editor allows you to define the percentage of the application budget that is available for each business service that the application provides. The sum of the percentages must be equal to or less than 100%.

Business Service Cost Allocation

	Function ▲	Cost Coverage %
1	Algorithmic Trading	15
2	Charting	12
3	Deposits Services	7
4	Order Validation	10
5	Pre-Trade Analytics	6
6	Routing the Order	15
7	Securities Transaction Processing	8
8	Trade Execution and Clearing	5
9	Trade Offering	5
10	Trade Pricing	6
11	Trade Risk Analytics	11

OK   Cancel

5) **Analyze the application costs of the owning organization**: The operational costs defined for the applications that an organization owns are used to calculate the operating expenses for the organization owning the application. An organization owns an application when it owns the ICT object to which the application belongs. This information is available in the *Operating Expenses Page View* for the owner organization.

6) In addition to the standard reports listed below, your enterprise may have configured reports that are specifically geared to the assessment and reporting needs of your enterprise. Your enterprise must purchase the relevant license to the expansion set Custom Reports. For more information about configuring reports, see the section *Configuring Reports* in the reference manual *Configuring Alfabet with Alfabet Expand*.

Analyzing Service Contract Costs for Organizations

Service costs are by definition OPEX costs. Therefore, reducing service costs allows you to reduce OPEX costs in your enterprise.

Some organizations may provide services that entail a service charge or fee. In Alfabet, service charges are defined for the use of the services provided by applications and ICT objects. The following steps are recommended to capture service contract costs for organizations in your enterprise:

1) **Create service contract templates**: This is carried out in the **Service Contracts Template Explorer**. When you create the service contract template, the following data should be defined:

- Each service contract template requires a unique name.

- Each service contract template requires a base price, which is the one-time fee for one-time services (for example, server installation, initial technical support, etc.).

- Each service contract template requires a usage price, which is the price calculated for one user's one-year usage. This is sometimes calculated through license prices per year if the license is a fixed license for one user.

- Each service contract template requires a one-time fee, which is the one-time fee for the first-time usage. This is for the users that start to use the service in the current year (for example, client installation).

- You should provide a description of the service contract template so that other users understand its purpose.

2) **Create service contracts for an organization based on the relevant service contract**: In the *Service Contracts Page View*, select the service contract template. The specified base price, usage price, and one-term fee are automatically applied to the new service contract. When you create the service contract, the following data should be defined:

   - Each service contract requires a unique name.

   - Each service contract requires a start and end date.

   - Each service contract requires a usage definition, which indicates the total number of users of the service for one contractual calendar year.

   - Each service contract requires a first time usage definition, which indicates the number of users who will begin to make use of the service in the contractual calendar year.

   - Each service contract must be explicitly calculated via the **Calculate** button in the **Service Contract** editor.

   - Each service contract must be set to active in order to indicate whether the selected organization is paying for that contract in the current year. Charges will only be aggregated from active service contracts.

3) **Analyze the services charges incurred by an organization**: In the *Business Service Charges Report*, examine the service charges that the selected organization is currently paying for. The price for the service charges of a service contract is calculated as follows: OneTimeFee * FirstTimeUsage + Base Price + UsagePrice * Usage. One Time Fee, Base Price and Usage Price are predefined in the corresponding service contract template. In the *Charge Back Report*, review the service charge that goes back to the selected organization based on the number of service contracts, the number of persons using the services defined in the service contract and also the object contribution defined in the service contract template that the service contracts are based on.

Chapter 4:   Cost Driver Analysis

Under budget constraints and a continuous flow of business demands, the enterprise must determine which investments and cost reduction measures best serve the business. The increased complexity of globally and organizationally distributed operations and associated IT makes it difficult to align IT plans and business needs.

Reducing operating expenses with the goal of improving efficiency, running business processes at lower cost, and outsourcing or reducing vendor costs is just as fraught with potential risks. Without the ability to capture costs appropriately to understand money spent on business services, business processes, and business capabilities, the enterprise cannot understand the actual costs incurred by the IT nor can it identify redundant or inefficient IT support of business processes and business capabilities, the consequences, postponement, or cancellation of system roll outs, or the potential business processes that may be candidates for an outsourcer who can deliver at lower cost.

Cost driver analysis helps enterprise architects, program managers, and IT controllers identify the costs of architecture elements and then aggregate these for individual business services, business processes, and business capabilities to understand exactly where costs are accruing. The clarity about which costs are driving the enterprise allow the organization to make good decisions without risk to the business' IT support and daily operations.

Cost driver analysis supports the enterprise to:

*   Make IT spending transparent to the business

*   Translate IT costs to business concepts

*   Align IT spending to the business' needs

The following information is available regarding the Cost Driver capability:

*   Methodology: Understanding Cost Driver Analysis

*   Analyzing Business Capability Costs

*   Analyzing Business Process Costs

*   Analyzing Organization Costs

ⓘ   Please note that a context-sensitive Help is available for each view available in the Cost Driver capability. You should refer to the Help if you require an explanation about the functionalities and information available in a specific view.

Methodology: Understanding Cost Driver Analysis

Alfabet provides various means to understand the costs driving the enterprise. In general, cost driver analysis is broken down into the following main tasks:

*   **Cost Structure Definition**: The definition of cost types and configuration of cost management functionalities is foundational to capturing the costs to operate the business and understanding. This is described in detail in the section Prerequisite: Configuration Requirements for Cost Management in the chapter OPEX Optimization.

- **Capture OPEX for IT Architecture**: In order to understand cost information and the relationships between IT and business processes, operational expenses (OPEX) must be assessed for the most significant architecture elements in the application architecture. Depending on the level of maturity of your enterprise, operating expenses may be captured via the definition of cost centers, ICT objects, or applications and their deployments. This is described in detail in the chapter OPEX Optimization.

> To capture projects and plan their budgets, you must have access to the Project Portfolio Governance capability, which is part of the IT Planning Advanced sales package.

- **Analyze Cost Aggregation of IT to Business**: Analyze the cost aggregation from IT artifacts to business artifacts including the costs of IT support for business capabilities, business processes, and organizations.

Analyzing Business Capability Costs

Business capabilities provides a methodological framework that focuses more explicitly on a high-level abstract description of what is done in an enterprise to meet its business objectives. Business functions describe the business activities relevant to the business capability and are realized by business services. Defining business functions helps to standardize business services on the demand-and-supply side by describing the business' IT requirements and offerings in a uniform, formalized, and comparable manner.

If your enterprise assesses its business objectives by means of business capabilities, you can assess the distribution of business service costs for the applications providing the business services associated with a selected business capability in the *Capability ICT Cost Report Page View* available in the *Business Capability Map Summary Page View*. The report shows the percentage of the application costs as well as the total costs of the application costs allotted to the business services. This information in conjunction with the *Business Support Map Report Page View* helps you to identify where money is being spent to support the enterprise's business capabilities and where costs can be cut to improve the financial efficiency of the enterprise. In addition, your enterprise may have configured reports that are specifically geared to the assessment and reporting needs of your enterprise. Your enterprise must purchase the relevant license to the expansion set Custom Reports. For more information about configuring reports, see the section *Configuring Reports* in the reference manual *Configuring Alfabet with Alfabet Expand*.

> To capture business capabilities, you must have access to the Business Capability Management capability, which is part of the Portfolio Management Basic sales package. The definition of business capabilities is described in the section *Business Capability Management* in the *Portfolio Management Basic* reference manual.

Analyzing Business Process Costs

Business processes represent the activities required to achieve a business objective. The following reports are available in Alfabet to help understand the money spent on the IT support of business processes:

> If business supports have been defined, you may configure the granularity of the calculation of business process costs based on the number of business services used by business supports. You can specify whether the cost calculation includes the costs of the business services that the business support uses or if the calculation of the business support costs for a business

process is based only on the costs of the business support. The calculation of business support costs must be configured by your solution designer in the configuration tool Alfabet Expand. For more information, see the section *Configuring Cost Management Capabilities* in the reference manual *Configuring Alfabet with Alfabet Expand*.

- *Business Service Costs Page View*: This report assesses the operational costs of business services provided by applications to the relevant business process. The report shows the percentage of the application costs as well as the total costs of the application costs allotted to the business services. The computation is based on the allocation of the business service costs to the business processes requesting those business services that are consumed by an organization associated with the selected business process by means of a business support.

- *Business Support Costs Page View*: This report assesses the business support costs for the selected business process and a selected organization that shares business supports with the selected business process. The computation of the costs is based on the service cost allocation for an application providing business services. Support costs for business services provided by an application are allocated to the business processes supported by business supports that provide those business services. The *Business Support Costs Hierarchy Page View* displays similar data but allows you to aggregate the costs along the cost types in the cost hierarchy or along the business processes in the business process hierarchy.

- *Cost Benchmarking Page View*: This report allows you to calculate and compare the costs for the selected business process as well as its sub-processes and helps to understand the cost efficiency of the business processes. In order to consider all aspects that are relevant for a comparison, you can select different evaluation types that you want to see indicators for. Business process costs can be compared for various organizations that conduct the same business process by means of different business supports or for different applications providing comparable business support for the same business process. The difference in costs may result from the number of business services that are provided by various business supports based on different applications as well as the various operational costs of different applications.

- In addition to the standard reports listed above, your enterprise may have configured reports that are specifically geared to the assessment and reporting needs of your enterprise. Your enterprise must purchase the relevant license to the expansion set Custom Reports. For more information about configuring reports, see the section *Configuring Reports* in the reference manual *Configuring Alfabet with Alfabet Expand*.

To capture business processes, you must have access to the Business Process Definition capability, which is part of the Enterprise Architecture Managements sales package. The definition of business process is described in the section *Business Process Definition* in the *Enterprise Architecture Management* reference manual.

Analyzing Organization Costs

Organization owners need cost transparency across their business units in order to make financially sound decisions about IT support. Alfabet provides this transparency with cost-driver analysis to highlight areas for cost optimization in the organization. Further, it supports efforts to optimize business unit investments and monitor IT budgets.

- *Operating Expenses Page View*: This report allows you to review the application costs for the owning organization: The application's operational costs are used to calculate the operating expenses for

the organization owning the application. An organization owns an application when it owns the ICT object to which the application belongs.

- *Business Support Costs Hierarchy Page View*: allows you to compute the business support costs for the selected organization and the business processes that share business supports with the selected organization. You can either aggregate the costs along the cost types in the cost hierarchy or along the organizations in the organization hierarchy.

- In addition to the standard reports listed above, your enterprise may have configured reports that are specifically geared to the assessment and reporting needs of your enterprise. Your enterprise must purchase the relevant license to the expansion set Custom Reports. For more information about configuring reports, see the section *Configuring Reports* in the reference manual *Configuring Alfabet with Alfabet Expand*.