



Portfolio Management Basic

Alfabet Reference Manual

Documentation Version Alfabet 10.15.0

Copyright © 2013 - 2022 Software AG, Darmstadt, Germany and/or Software AG USA Inc., Reston, VA, USA, and/or its subsidiaries and or/its affiliates and/or their licensors.

Use, reproduction, transfer, publication or disclosure is prohibited except as specifically provided for in your License Agreement with Software AG.

The name Software AG and all Software AG product names are either trademarks or registered trademarks of Software AG and/or Software AG USA Inc. and/or its subsidiaries and/or its affiliates and/or their licensors. Other company and product names mentioned herein may be trademarks of their respective owners.

This software may include portions of third-party products. For third-party copyright notices, license terms, additional rights or restrictions, please refer to "License Texts, Copyright Notices and Disclaimers of Third Party Products". For certain specific third-party license restrictions, please refer to section E of the Legal Notices available under "License Terms and Conditions for Use of Software AG Products / Copyright and Trademark Notices of Software AG Products". These documents are part of the product documentation, located at <http://softwareag.com/licenses> and/or in the root installation directory of the licensed product(s).

Software AG products provide functionality with respect to processing of personal data according to the EU General Data Protection Regulation (GDPR). Where applicable, appropriate steps are documented in the respective administration documentation.

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.
<i>Italics</i>	Used for emphasis, titles of chapters and manuals. this Example: see the <i>Administration</i> 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 Basic	5
Chapter 2: Application Portfolio Governance	6
Methodology: Understanding Application Portfolio Governance	7
Prerequisite: Configuring Evaluation Data for Application Portfolio Assessment	9
Configuring Evaluation Types	11
Configuring Aspect Evaluation Types	12
Configuring Analytics to Understand the Application Evaluation	13
Creating and Managing Application Portfolios	14
Evaluating the Applications and Analyzing the Application Portfolios	17
Chapter 3: Information Portfolio Governance	19
Methodology: Understanding Information Portfolio Governance	19
Prerequisites: Configuring Evaluation Data for Information Portfolio Governance	20
Capturing Business Data Usage	20
Analyzing Business Data Usage	22
Implementing Proposed Information Flows to Manage the Information Architecture	23
Chapter 4: Technology Portfolio Governance	25
Methodology: Understanding Technology Portfolio Governance	26
Prerequisite: Configuring Evaluation Data for Components	27
Importing Components from the Technopedia® Repository	30
Creating, Managing, and Evaluating Component Portfolios	31
Managing Component and Standard Platform Compliance via Component Catalogs	32
Implementing Proposed Local Components to Manage Technology Upgrades and Lifecycles	35
Prerequisite: Configuring the Implementation of Proposed Local Components	35
Managing Device Details and Device Groups	36
Prerequisites: Configuring the Capture of Device Details	37
Chapter 5: Business Capability Management	38
Documenting the Enterprise's Functional Domains	39
Methodology: Understanding Domain Models	40
Prerequisite: Configuring the Domain Management Capability	43
Understanding Governance and Responsibility for Domain Models	45
Documenting and Defining the Domain Model	46
Assigning Architecture Objects to a Domain	47
Defining Business Functions and Business Services	48
Defining the Operational Business Support for a Domain	50
Documenting the Service Architecture for a Business Process	51
Analyzing the Domain Model	51
Documenting the Enterprise's Business Capabilities	52
Prerequisite: Configuring the Business Capability Management Capability	54
Defining Business Capabilities and Creating a Business Capability Map	55
Evaluating the Business Capabilities	58
Analyzing the Enterprise's Business Capabilities	59
Capturing Business Appraisals in the Enterprise	60

Chapter 1: Introduction to Portfolio Management Basic

The Portfolio Management Basic sales package allows you to understand, standardize, and govern your company's complex IT systems. The capabilities available in the Portfolio Management Basic sales package enable you to evaluate how well the IT supports the enterprise's business requirements. It enables you to evaluate and identify which business capabilities are core and where improvement in one or the other of the portfolios is needed. Portfolio governance capabilities enable the organization to neatly organize the information needed to optimize portfolios, improve agility and set the pace for change. Tightly integrated portfolios deliver impact analyses to reduce planning errors and improve synergies between the portfolios. The Portfolio Management Basic package is used by portfolio managers, business analysts and enterprise architects to:

- perform rapid impact analysis on the application, technology and information portfolios resulting from business events such as acquisitions, divestitures, product launches, and entry into new markets
- support IT management processes that require reliable information on applications, technologies and information flows
- choose, define and enforce technology standards
- identify technology innovations that can provide business benefit to the enterprise
- assess critical business aspects according to business capabilities and evaluate the related IT support

The Portfolio Management Basic sales package enables a wide variety of analyses - both standard and customized - to evaluate your enterprise's IT landscape. Most of these typically require minimal configuration in order to be implemented in your enterprise's solution. As a result, most evaluations and reports will be tailored to assess the data in ways that are relevant for your enterprise. Once evaluation types and indicator types, for example, have been configured, they can be implemented to analyze and help you to understand your enterprise's application, information, technology and business portfolios. For detailed information about how to create and configure evaluation types and indicator types in the **Configuration** module, see the section *Configuring Evaluation Types* in the reference manual *Configuring Evaluation and Reference Data in Alfabet*. For more information about creating configured reports, see the section *Configuring Reports* in the reference manual *Configuring Alfabet with Alfabet Expand*.

The following information is available:

- [Introduction to Portfolio Management Basic](#)
- [Application Portfolio Governance](#)
- [Information Portfolio Governance](#)
- [Technology Portfolio Governance](#)
- [Business Capability Management](#)

Chapter 2: Application Portfolio Governance

Because running applications usually constitute the largest part of IT spending, application portfolio governance is crucial to containing costs and to ensuring that the IT support to the business is consistent and reliable. Therefore, an application portfolio governance program is a necessity to control and manage the application architecture. The goals informing application portfolio governance include:

- Improve the alignment of applications with the business strategy and ensure the applications support the business capabilities
- Reduce the applications' exposure to risks
- Increase the agility and flexibility of the application portfolio to address the enterprise's changing business needs
- Reduce the running costs of the applications

Whether such goals are relevant to all application portfolios will depend on the needs of the enterprise and the defined focus of the various portfolios.

Alfabet provides an Application Portfolio Governance capability that allows you to evaluate and understand the application architecture and manage the complexity of your applications. This capability allows you to evaluate and analyze applications in the context of various application portfolios in order to understand the technical and information architectures relevant to the applications, the use of the applications by the business as well as application costs, failure rates, and risks. The assessment of applications and application portfolios allows you to understand the usage, performance, and criticality of an application in the IT landscape. Attaining answers to the following questions is critical to the maintenance of a healthy and cost-effective architecture as well as planning future operating models:

- Which applications should be retired?
- Which applications should be invested in?
- Which applications can be consolidated or optimized?
- Which applications are to be tolerated in the landscape but need observation?
- Which applications have risks that should be mitigated?
- Which applications need to have their lifecycles extended or reduced?



Before you can begin to evaluate your company's application portfolios, information on the organization's application landscape must already be captured and maintained. The activities around information gathering are done at the level of individual applications and are typically coordinated by an application owner or other responsible staff. Ideally, all relevant applications in your enterprise have already been captured in the inventory. Information about the applications, such as the business data that is transferred by them, the technical context required to run the application, the business processes that the application supports, and the functional domains that own the application should also be documented and up-to-date. For detailed information about capturing application data, see the reference manual *Enterprise Architecture Management*.

The following information is available regarding the Application Portfolio Governance capability:

- [Methodology: Understanding Application Portfolio Governance](#)
- [Prerequisite: Configuring Evaluation Data for Application Portfolio Assessment](#)

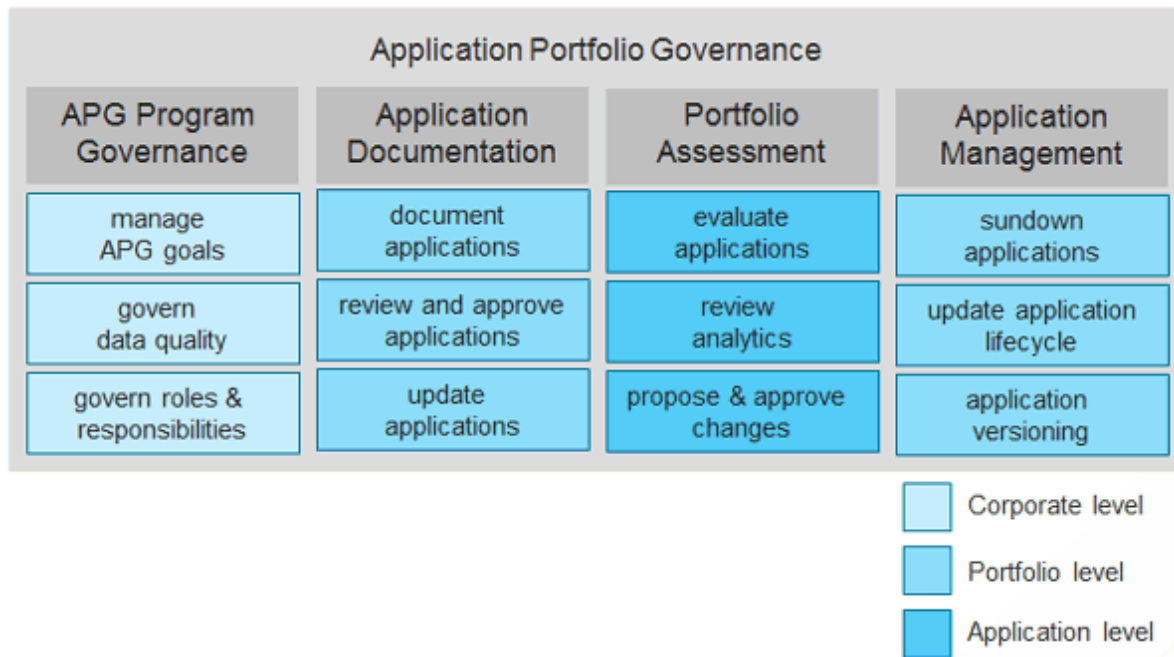
- [Configuring Evaluation Types](#)
- [Configuring Aspect Evaluation Types](#)
- [Configuring Analytics to Understand the Application Evaluation](#)
- [Creating and Managing Application Portfolios](#)
- [Evaluating the Applications and Analyzing the Application Portfolios](#)



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

Methodology: Understanding Application Portfolio Governance

In general, application portfolio governance is broken down into the following main tasks:



- **Govern APG Program:** The governance of the portfolios typically occurs on the corporate level and includes conceptualizing and managing the application portfolios, ensuring data quality, and establishing the roles and responsibilities for the application portfolios.
- **Capture Application Architecture:** The gathering of information is fundamental to application portfolio governance. The applications in the IT architecture must be documented and bundled in relevant application groups. To ensure that this occurs, it is recommended that a process is implemented in the enterprise whereby a release status indicates the quality of information about the application. For example, an application with the status "Approved" would indicate to the organization that the application is sufficiently described and is ready for evaluation. An application that is "Approved" could require the following information to be considered complete:
 - Name, description, version

- Lifecycle and release status definition
- Business and functional ownership
- Role assignments that describes the functional relationship or responsibility that a user or organization has to the application
- Business processes and organizations that the application provides operational support to
- **Assess Application Portfolios:** The assessment of the portfolio requires that applications are bundled into application groups that logically structure the applications to be assessed. The evaluation occurs based on configured indicators used to evaluate the applications as well as standard and configured reports. The choice of KPIs and analytics will depend on the goals for the portfolio assessment. For example, if the goal is to reduce the number of applications in the IT landscape, then it is necessary to understand how much an application is used in the business and the amount of overlap there is between applications. In this case then, evaluation types must be configured to capture information about application use and application overlap. The goals of the portfolio assessment should be carefully considered when conceptualizing and configuring the KPIs to capture for applications and application groups.
- **Plan and Manage Application Architecture:** The management of the applications focuses on managing the outcomes of the evaluation and, for example, coordinating changes to the application lifecycle, versioning applications, and potentially sunsetting applications. The tasks of managing application lifecycles and versioning is described in the context of capturing application data in the chapter *Application Architecture Definition* in the reference manual *Enterprise Architecture Management*.

Many different roles are required in the enterprise to achieve these goals. Although roles may vary among different companies, the table below represents an overview of the typical types of roles involved in the application portfolio governance process:

Role Name	Responsibilities
Program Manager	Responsible for the overall APG process: <ul style="list-style-type: none"> • Defines and coordinates overall goals of APG • Defines governance framework including monitoring of data quality.
Domain Architect	Responsible for functional domains and functional governance of applications providing support to the domains: <ul style="list-style-type: none"> • Analyzes functional overlaps and gaps • Contributes to the application strategy for a functional domain
Product Manager	Responsible business-context of application and for IT product roadmap: <ul style="list-style-type: none"> • Analyzes requirements and proposes new application versions • Documents basic application information regarding business supports and supported business processes and organizations

Role Name	Responsibilities
	<ul style="list-style-type: none"> Supports evaluation of application from a business perspective
Business Owner	Responsible for budget: <ul style="list-style-type: none"> Monitors application performance via reports Approves roadmaps and budgets Reviews business perspective assessments
Business Contact	Responsible as subject matter expert for business needs and use of the application:
Program Manager	Responsible for IT-context of application: <ul style="list-style-type: none"> Documents information architecture and technology platform relevant to application Assesses application alignment with IT strategy Assesses development effort/cost

Prerequisite: Configuring Evaluation Data for Application Portfolio Assessment

Once the goals of the portfolio assessment have been conceptualized and applications have been assigned to the application groups, the details of the evaluation must be specified. This includes the configuration of evaluation types and indicator types. It is useful to consider the various perspectives that are necessary to evaluate the application landscape and the questions that may need to be answered for each perspective:

- Business perspective: The business perspective aims to understand the business usage of applications, how critical they are to the business, and what business requirements will impact the applications. In this case, for example, indicators could be captured that answer the following kinds of questions:
 - Which applications are most/least critical to business operations?
 - Which applications provide the most/least value to business?
 - Which applications have the highest number of users?
 - Which applications is the business most/least satisfied with?
 - Which applications must be agile in order to support changing business needs?
 - Which applications are crucial to future business strategies?

- Functional perspective: The functional perspective aims to understand the functional domains or business capabilities that the application supports. In this case, for example, indicators could be captured that answer the following kinds of questions:
 - How large is the functional scope of an application?
 - Which applications have functional overlap and could be consolidated?
 - Which applications have unused functionality?
 - Which applications are not well aligned to the business capability they support?
- Technology perspective: The technology perspective aims to align the technology platform of the application with the enterprise's technology standards and strategy. In this case, for example, indicators could be captured that answer the following kinds of questions:
 - Which applications should be migrated due to aging technology usage?
 - Which applications are not in line with current technology standards?
 - Which applications are easy to maintain?
- Cost perspective: The cost perspective aims to understand the cost drivers in the application architecture and align spending with business priorities. In this case, for example, indicators could be captured that answer the following kinds of questions:
 - What are the operating costs of applications?
 - What are the operational costs per functional domain?
- Operations perspective: The cost perspective aims to understand the operational performance of the applications in order to identify areas that need attention. In this case, for example, indicators could be captured that answer the following kinds of questions:
 - Which applications have the highest incidence rates?
 - Which applications fail most frequently?
 - Which applications do not meet recovery time objectives?
- Risk and compliance perspective: The risk and compliance perspectives aims to understand which applications have risk and compliance issues in order to ensure that threats are identified and risks mitigated. These issues are covered in the Application Risk Management capability which is available in the IT, Governance, Risk and Compliance sales package. For more information, see the reference manual *IT Governance, Risk and Compliance*.

The following provides an overview of the various mechanisms to evaluate and analyze data in order to assess applications in the context of application portfolio governance:

- [Configuring Evaluation Types](#)
- [Configuring Aspect Evaluation Types](#)
- [Configuring Analytics to Understand the Application Evaluation](#)

Configuring Evaluation Types

Performance indicators are an important means to understand various aspects of an application in the context of the IT architecture and business landscape. In the *Evaluation Page View*, users with access permissions to an application can enter values for customer-defined indicator types that an authorized user in your enterprise has configured in the **Evaluations and Portfolios** functionality. The defined indicators are be used in various standard or configured reports available in Alfabet. Indicators displayed in Alfabet may be either:

- manually entered by a user
- selected from a predefined set of values by a user
- calculated based on other indicators or data in Alfabet according to computation rules defined by your enterprise
- calculated based on customer-specific code provided by Software AG Support

To evaluate and understand the application portfolio, evaluation types can be configured for the object classes **Application**, **Application Group**, **ICT Object**, and **ICT Object Group**. The indicators that are to be evaluated will depend on the goals of the application portfolio governance project. For example, a cost assessment might require indicators that identify applications that constitute 70% of the costs, the applications that constitute 20% of the costs, and the applications that constitute 10% of the costs. A detailed analysis of the applications that make up 70% of the costs might result in the opportunity to reduce obsolete or redundant applications. A subsequent assessment focused on the relevance of an application for the business might include indicators to evaluate the agility of an application, its criticality to the business process it supports, its contribution to the business, the satisfaction with the functionality of the application, the criticality of the application to the business as a whole, and the usability of the application.

Evaluation and indicator types can be configured by a user with access permissions to the **Configuration** module available in the Alfabet interface. Indicator types are bundled in evaluation types and can be reused for a multitude of classes, if necessary. Indicators can also be automatically generated by specifying the calculation (such as sum, minimum, maximum, or average) of multiple indicators or the aggregation of multiple indicators for objects in an object hierarchy. In the case of a computation rule, one or more indicators are bundled in evaluation types that can be aggregated to provide a single value. Depending on your enterprise's solution configuration, users might manually enter values for each indicator type or select a value from a preconfigured set of values. Some evaluation types may have values that are generated by means of configured computation rules.

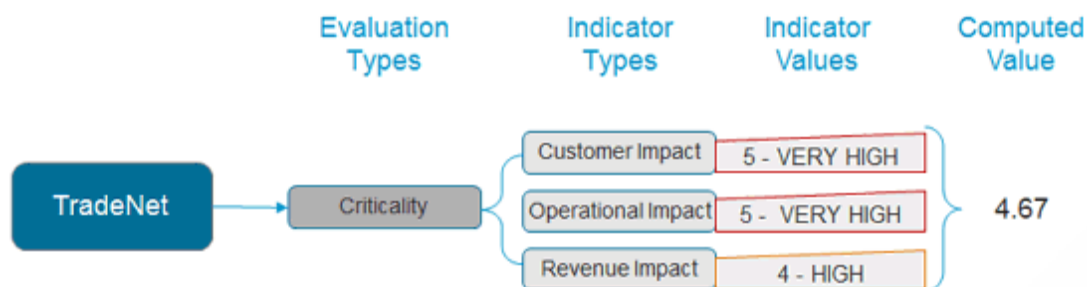


FIGURE: Evaluation of the criticality of the application Trade*Net

To assess the criticality of an application in the enterprise, for example, an enterprise could specify an evaluation type Criticality with indicator types that evaluate the impact that would be experienced if the application were out of service for one day. In the example above, the three indicator types Customer Impact,

Operational Impact, and Revenue Impact are defined by the user. The average of the three indicators is computed to provide one overall indicator expressing the application's criticality.

To extend this example, the enterprise may want to understand the criticality of applications in all application groups subordinate to a parent application group as well as the criticality of all of the application groups subordinate to the parent application group. This would require two indicators, one indicator that computes the criticality of applications based on the indicators Customer Impact, Operational Impact, and Revenue Impact (shown in the figure above) and the other indicator that computes the criticality of the application groups in an application group hierarchy.

For detailed information about how to create and configure evaluation types and indicator types in the **Configuration** module, see the section *Configuring Evaluation Types* in the reference manual *Configuring Evaluation and Reference Data in Alfabet*.

Configuring Aspect Evaluation Types

Another method to evaluate applications in Alfabet is to use aspect evaluations. Because an application may be a member of several different application groups, different qualitative assessments of the same application using the same evaluation criteria are possible for each application group that the application is assigned to. Aspect evaluations allow the indicators to be differently defined for applications in the context of different application groups.

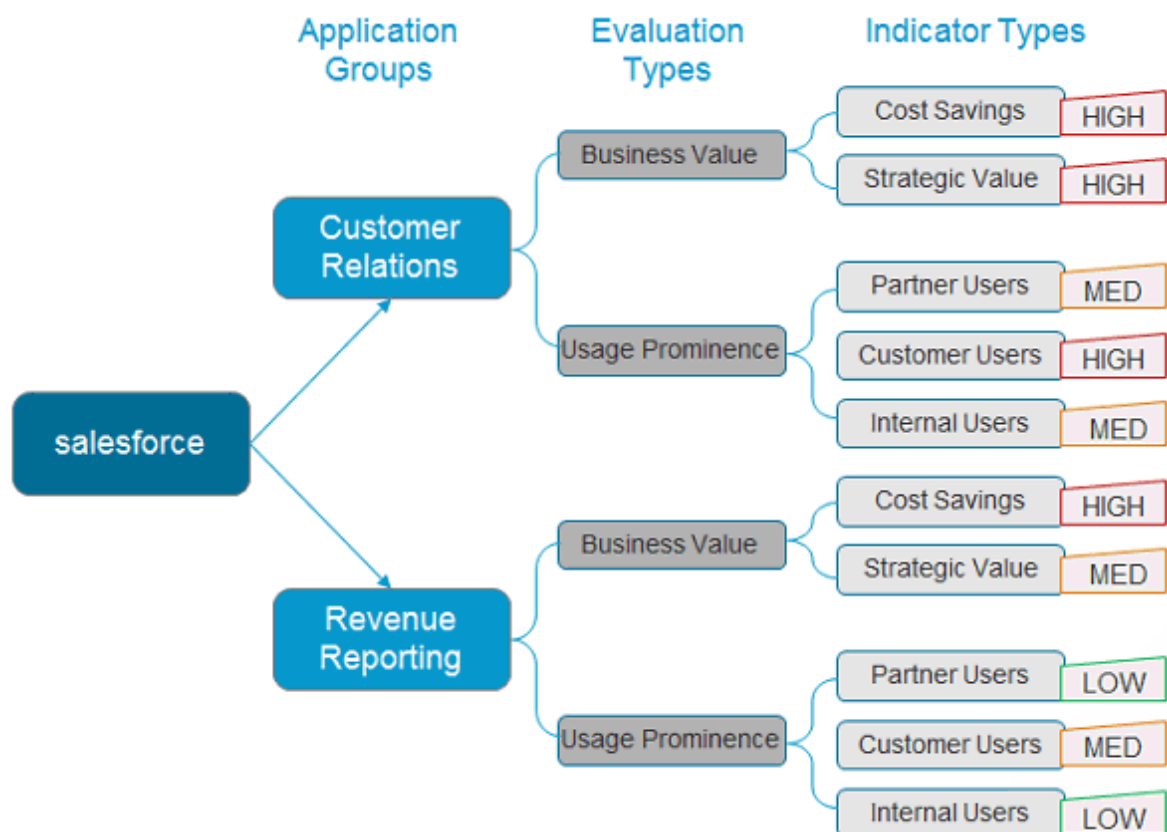


FIGURE: Evaluation of an application from the perspective of different application groups

For example, the application salesforce.com may be relevant for different purposes in the enterprise architecture. The application may be suitable to varying degrees both as a customer relationship management tool and a revenue reporting tool. An aspect evaluation would allow you to examine the application from the

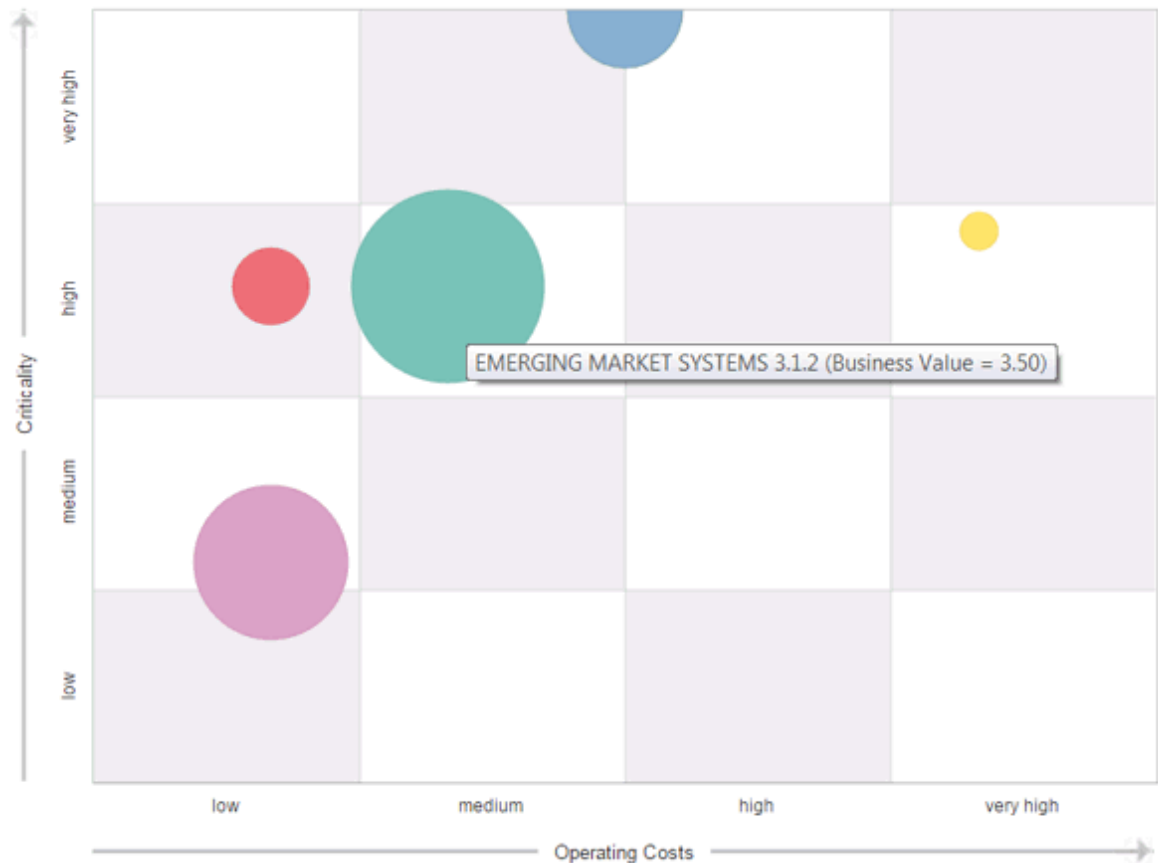
perspective of the application group Customer Relations and the application group Revenue Reporting based on specific aspect evaluation types, such as Business Value or Usage Prominence.

Like conventional evaluation types, aspect evaluation type comprise one or more indicator types that are used to evaluate the applications. Aspect evaluations can also be analyzed in the context of aspect portfolio reports. For more information about configuring aspect evaluations as well as aspect portfolios, see the section *Configuring Aspect Evaluations for the Classes Application and Component* in the reference manual *Configuring Evaluation and Reference Data in Alfabet*.

Configuring Analytics to Understand the Application Evaluation

A wide variety of evaluation mechanisms can be configured in order to assess the application data based on the needs of your enterprise.

- **Prioritization schemes:** In addition to configuring evaluation types that bundle indicator types, a user with access permissions to the **Configuration** module can also configure prioritization schemes, which bundle one or more sets of evaluation types. A prioritization scheme is a weighted composite of a set of evaluation types and is typically used to determine a prioritized ranking for a set of objects. For example, for a prioritization scheme Business Relevance, the evaluation type Architecture Impact may be have a 20% rating, the evaluation types Business Value and Strategic value may each have 40% weightings. When the user selects a prioritization scheme in a report, the weighting of the evaluation types is automatically applied to produce a value. A prioritization scheme is typically used to define an axis in a portfolio report.
- **IT portfolios:** A portfolio report is a bubble chart that represents the relative performance of a set of objects that belong to the same object class in two or three independent dimensions of measurement. A portfolio report may display either a conventional X-, Y-, and power-axis or a BCG quadrant. Typical portfolio reports relevant to application portfolio governance might display a group of applications that are owned by a specific organization, used by a particular business process, or assigned to an application group. In order to work with portfolio reports in Alfabet, IT portfolios must be configured by a user with access permissions to the **Configuration** module.



The example above shows a configured IT portfolio Operating Costs vs. Criticality displayed in the *Application Portfolio Page View* for the application group Trade Entry. The portfolio shows all applications in the application group as bubbles of varying sizes. The X-axis represents the evaluation type Operating Costs, the Y-axis displays the evaluation type Criticality, and the Z-axis (the bubble size) represents the evaluation type Business Value. The tooltip displays the application's name and version as well as the Z-axis value. The axes of the IT portfolio may represent either a single indicator type, an evaluation type bundling indicator types, or a prioritization scheme bundling evaluation types. For more information about configuring portfolios, see the section *Configuring Portfolios* in the reference manual *Configuring Evaluation and Reference Data in Alfabet*.

- Configured reports:** Depending on the configuration of your Alfabet solution, many of the reports that you see may be configured reports that have been created by your solution designer in order to provide a customized data analysis that is not available in the standard Alfabet product. Reports may be configured to display indicators in tables, matrices, portfolios, or Gantt charts. Other graphic visualizations may also be displayed in configured reports including, for example, radar charts, line charts, pie charts, tree map reports, layered diagrams or cluster map reports. For more information about creating configured reports, see the section *Configuring Reports* in the reference manual *Configuring Alfabet with Alfabet Expand*.

Creating and Managing Application Portfolios

Application portfolios are defined in Alfabet via application groups that bundle applications, or on a higher and more abstract level, via ICT object groups that bundle ICT objects that own applications. Each

application or ICT object can be assigned to multiple groups. The applications should be bundled in relevant application groups that are relevant to the goals for the portfolio assessment process. For example, applications could be logically structured in application groups according to:

- the high-level business processes that the applications support.
- the organizations that the applications support
- the organizations that are responsible for the operation and maintenance of the applications
- the technology that the applications use
- or any other ad-hoc assessments of segments of the IT landscape

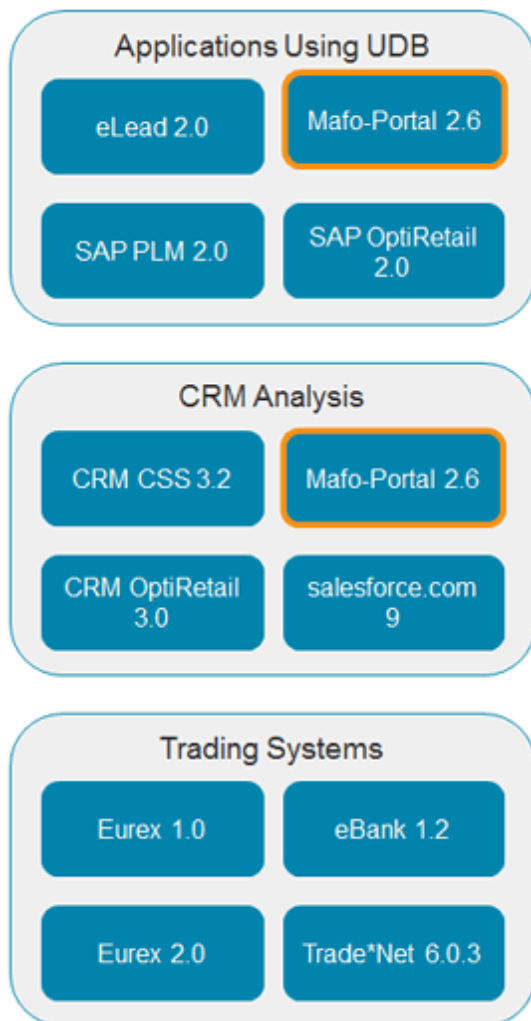


FIGURE: Application Mafo-Portal 2.6 is assigned to two application groups

For example, a banking enterprise wants to create application groups according to the company's organizational structure in order to understand costs, alignment, risks, etc. per organizational unit and therefore defines the application groups Market Data, Trade Entry, Front Office, Back Office, Customer Relations, etc.

In addition to looking at applications per organization, the enterprise also wants to evaluate applications based on the technologies used in the enterprise in order to understand which applications should be sun-downed and which should be further maintained. In this case, for example, another application group

hierarchy could be defined with applications groups such as Applications Using UDB, CRM Analysis, Trading Systems, etc. You can decide on the number of application group hierarchies and levels in each hierarchy.

Application groups are created at the root node of the *Applications by Group Explorer*. Subordinate groups are created in the *Sub-Groups Page View* of the parent application group. This explorer is especially useful for portfolio managers due to the various reporting business capabilities available for application portfolios. A number of governance concepts are implemented for application groups:

- **Authorized User:** Each application group has an authorized user. An authorized user has primary responsibility for the application group 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 an application group will have Read/Write access permissions to the application.
- **Roles:** A role defines the functional relationship or responsibility that a user or organization has to an application group (for example, the Risk Manager or Application Architect). Various roles may be required in order to provide input to the application portfolio from various perspectives. Roles describe responsibilities but they do not authorize access permissions to the application group in Alfabet.
- **Mandates:** Application groups may be managed in a federated architecture. By means of mandates, it is possible to specify the visibility of individual application groups in the Alfabet interface for specific users. The mandate capability may be activated or deactivated for the entire capability as well as explicitly activated or deactivated for specific application group stereotypes.
- **Object Class Stereotypes:** Object class stereotypes may be configured by your solution designer for the object class Application Group. This allows for a different governance approach between different kinds of application groups. If object class stereotypes are configured for the object class Application Group, each stereotype may capture a specified set of attributes, reference data, and class configurations as well as implement a different governance approach.

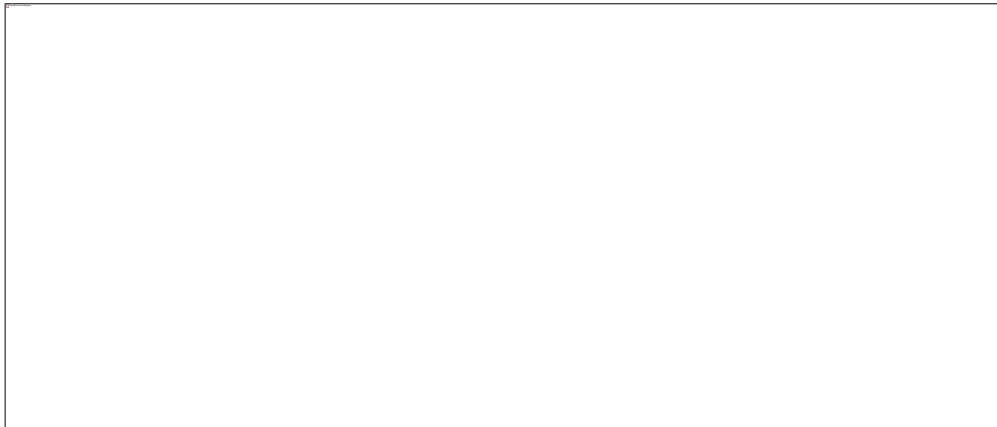


FIGURE: ICT object group Trading with its assigned ICT objects and their assigned applications

ICT object groups are typically used by budget owners in order to understand the costs of a particular set of applications. Understanding portfolios based on ICT objects is advantageous because the IT architect and planner must not initially commit him/herself to a certain version of an application. The ICT object can be associated with multiple applications as well as the components, devices, and standard platforms that the application uses. As such, the application costs including the technical infrastructure can be captured in an ICT object. Later, at the stage of detailed planning, the ICT object can be replaced by a specific concrete application version. By examining the application landscape from the perspective of the ICT object group, costs for planned applications can be understood in the later phases of master planning and

strategic planning. The task of capturing and defining ICT objects is described in the section *Capturing ICT Objects to Understand IT Costs* in the reference manual *Enterprise Architecture Management*.

ICT object groups are created at the root node of the *ICT Objects by Group Explorer*. Here ICT objects can be assigned to ICT object groups. Furthermore, ICT object owners can assign applications to their ICT objects in order to track and analyze application costs.

- **Authorized User:** Each ICT object group has an authorized user. An authorized user has primary responsibility for the ICT object group 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 an ICT object group will have Read/Write access permissions to the ICT object group.
- **Roles:** A role defines the functional relationship or responsibility that a user or organization has to an ICT object group (for example, a Program Manager or Budget Owner). Various roles may be required in order to provide input to the application portfolio from various perspectives. Roles describe responsibilities but they do not authorize access permissions to the ICT object group in Alfabet.
- **Mandates:** ICT object groups may be managed in a federated architecture. By means of mandates, it is possible to specify the visibility of individual ICT object groups in the Alfabet interface for specific users. Mandates are typically configured if ICT object group stereotypes are configured. The mandate capability may be activated or deactivated for the entire capability as well as explicitly activated or deactivated for specific ICT object group stereotypes.
- **Object Class Stereotypes:** Object class stereotypes may be configured by your solution designer for the object class ICT Object Group. This allows for a different governance approach between different kinds of ICT object groups. If object class stereotypes are configured for the object class ICT Object Group, each stereotype may capture a specified set of attributes, reference data, and class configurations as well as implement a different governance approach.



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*.

Evaluating the Applications and Analyzing the Application Portfolios

Once the assessment goals have been determined and the relevant evaluation types and indicator types have been configured for the application portfolio assessment, the actual evaluation process can begin. Typically, each application will be evaluated by its authorized user or a member of the authorized user group in the *Evaluation Page View* in the application's object profile. The view will list all evaluation types and their indicator types that have been assigned to the object class Application. Depending on the configuration of the indicator type, the user may be required to enter a numerical indicator or select a value in the drop-down list. It may be required by your enterprise that you explicitly define indicators for an application group as well. The *Evaluation Page View* is also available in the object profile of an application group and allows application groups as a whole to be evaluated.

Once the applications have been evaluated, you can look at various reports that allow you to analyze the application portfolio from a variety of perspectives. 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*. The following standard reports are available:

- The *Application Evaluation Report Page View* is available on the root node of the *Applications by Group Explorer* and displays all applications in all application groups and their aggregated indicators for a selected prioritization scheme. The *Application Evaluation Report Page View* is also available for a selected application group and focuses on the aggregated indicators for all applications in the selected application group according to a selected prioritization scheme.
- The *Application Lifecycle Page View* displays the lifecycles and lifecycles phases of the applications and their application variants assigned to the selected application group as well as the lifecycles of the ICT object that owns the application. In this way, you can review the application lifecycles and identify and manage any conflicts in these lifecycles.
- The *Application Portfolio Page View* allows you to analyze all applications in a selected application portfolio based on configured IT portfolios and the *Sub-Group Portfolio Report* allows you to understand the assessment of all application groups subordinate to a selected application group.
- The *Aspect Portfolio Page View* allows you to examine qualitative information about the applications in the selected application group based on various configured aspects.
- The *Business Comparison Report Page View* allows you to understand the business overlap between two application groups in terms of the business services that are offered, the business data that are transferred, and the business processes that are supported.

Chapter 3: Information Portfolio Governance

Information Portfolio Governance supports organizations to understand who owns business data, who uses the business data and where this business data is being processed. It focuses on understanding the information flows and dependencies between applications, including such issues as the quality and criticality of the information flows for business continuity. Using CRUD logic, your enterprise can document the business data usage, which describes whether business data is created, read, update, deleted, or processed in the context of an application, component, information flow, or business service as well as whether business data is an input and/or output of an architecture element. A standard CRUD matrix allows users to view and identify conflicts in business data usage in order to identify inconsistencies and redundancies in the enterprise's application landscape and provides valuable input into the potential consolidation of applications.



The perspective regarding threats and risk to business data as well as planning and implementing risk mitigations is covered in the Information Risk Management capability which is available in the IT, Governance, Risk and Compliance sales package. For more information, see the reference manual *IT Governance, Risk and Compliance*.

The following information is available:

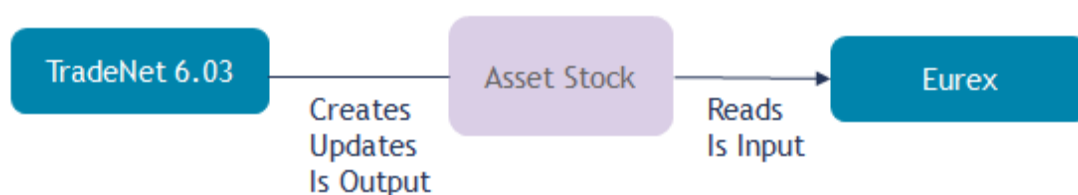
- [Methodology: Understanding Information Portfolio Governance](#)
- [Prerequisites: Configuring Evaluation Data for Information Portfolio Governance](#)
- [Capturing Business Data Usage](#)
- [Analyzing Business Data Usage](#)
- [Implementing Proposed Information Flows to Manage the Information Architecture](#)



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

Methodology: Understanding Information Portfolio Governance

Business data usage describes whether business data is created, read, update, deleted, or processed in the context of an application, component, information flow, or business service as well as whether business data is an input and/or output of an architecture element. Business data usage is an important means to identify inconsistencies and redundancies in an enterprise's application landscape. A standard CRUD matrix allows users to view and identify conflicts in business data usage.



The application TradeNet creates and updates the business data Asset Stock, which is used by the outgoing information flow of the application TradeNet. The application Eurex reads the business data Asset Stock, which is used by the incoming information flow of the application Eurex.

Prerequisites: Configuring Evaluation Data for Information Portfolio Governance

Performance indicators are an additional means to understand various aspects of the business data and their information flows in the context of the IT architecture and business landscape. In the *Evaluation Page View*, users with access permissions to a business data or information flow, for example, can enter values for customer-defined indicator types that an authorized user in your enterprise has configured in the **Evaluations and Portfolios** functionality. The defined indicators are be used in various standard or configured reports available in Alfabet. Indicators displayed in Alfabet may be either:

- manually entered by a user
- selected from a predefined set of values by a user
- calculated based on other indicators according to computation rules defined by your enterprise
- calculated based on customer-specific code provided by Software AG Support
- imported via an interfacing system such as ADIF

Evaluation types can be configured for the object classes **Business Object**, **Business Data**, **Business Data Usage**, and **Information Flow** in order to support information portfolio governance. The indicators that are to be evaluated will depend on the goals of the information portfolio governance project. For example, to understand the quality of information flows, an evaluation type Information Flow Quality could be configured to include the indicator types Interface Criticality, Interface Reliability, and Technical Compliance. Or to gain a better understanding about individual business data, an evaluation type Data Assessment could be configured to include the indicator types Confidentiality, Availability, Reliability, and Integrity.

For detailed information about how to create and configure evaluation types and indicator types in the **Configuration** module, see the section *Configuring Evaluation Types* in the reference manual *Configuring Evaluation and Reference Data in Alfabet*.

Capturing Business Data Usage

Before you can begin to evaluate your company's information architecture, your enterprise's applications, their information flows, and the business data they transfer must already be captured, maintained, and up-to-date. Similar to applications, components may also have information flows that transfer business data. In addition, business services may also use business data to fulfill a business process. In this case, the business services are provided by applications or components that operate on business data. For detailed information about capturing information flows and business data as well as the applications, components, or business services operating on the business data, see the reference manual *Enterprise Architecture Management*.

Once the business data that the applications, components, and business services use have been defined, you can define in more detail how the business data are used. This is captured by means of the notion of CRUD (Create, Read, Update, Delete). CRUD has emerged in the industry as an appropriate means to identify the specific interaction between an application (or component or business service) and a specific business data. The CRUD logic implemented in Alfabet is as follows:

Business Data/Business Object Usage ?

Asset.Stock v. 1.5

☒ Create (C) ☐ Is Input (I)

☐ Read (R) ☒ Is Output (O)

☒ Update (U)

☐ Delete (D)

☐ Processing (P)

Data Retention Policy

Asset, 3 Years 🔍

Usage Comments

ENU OK Cancel

- C = Create: The business data/business object is created by the application, component, or business process.
- R = Read: The business data/business object is read by the application, component, or business process.
- U = Update: The business data/business object is updated by the application, component, or business process. Please note that this does not include the initial creation of the object.
- D = Delete: The business data/business object is deleted by the application, component, or business process.
- P = Processing: The application, component, or business process processes data pertaining to the business data/business object. Processing indicates that the business data is handled in the context of the relevant object without the data being altered, created or deleted. The data is typically kept locally so that a desynchronization results between the processed business data and the system(s) of records for the business data. Typically, it is not the raw data that is processed but some aggregation or abstraction of the business data (for example, data in an operational data store or an OLAP-based reporting solution).

- **I = Is Input:** The business data/business object is used by the incoming information flow of the application, component, or business process.
- **O = Is Output:** The business data/business object is used by the outgoing information flow of the application, component, or business process.

The business data usage is defined in the *Business Data Page View* of the application, component, or business service using the data. Although you can define all combinations of CRUDP and Input/Output, it is recommended that you make logical definitions. For example, the definition UD (Update and Delete without Create) only makes sense in the case of an imported business data. In this case, the Input checkbox should also be selected. Once the business data usage has been defined for all relevant applications, components, and/or business services, you can analyze the business data usage and search for redundancies, conflicts, and inconsistencies in the *CRUD Matrix*.

The business data usage that you define here is also displayed in the *CRUD Matrix Page View* for an application.

Analyzing Business Data Usage

Large organizations face the challenge that oftentimes the same business data is maintained in different applications. This is a major source for inconsistency, error, and intransigence. As a result, enterprise architects need to understand where specific business data is maintained and used. The notion of CRUD (Create, Read, Update, Delete) has emerged in the industry as an appropriate means to identify the specific interaction between an application and a specific business data. This notion is extendable to other architecture objects dealing with business data – in particular business services, information flows, and components.

In order to make this information available in a clear and easy-to-manage fashion, Alfabet provides CRUD matrices. A CRUD matrix is a well-established means to represent business data usage for applications, components, or business services. In a CRUD matrix, architecture elements (applications, components, and business services) represent one dimension of the matrix and business data represent the other dimension of the matrix. The corresponding cells indicate how the architecture element and the business data are related to one another (if at all). A variety of analyses using a CRUD matrix help the user to identify inconsistencies and redundancies in the way business data are used in the IT landscape. A standard CRUD matrix supports you to do the following:

- View and identify conflicts in business data usage.
- Identify equal or similar data objects that are processed in different applications and are created or updated in different systems
- Identify which systems are responsible for which data objects to simplify the data search and data analysis

Application: Trade*Net 6.0.3 [APP_COM_CRUDMatrix]

CRUD Matrix

Object State

Active ☐ Transpose☐ Consider Business Services

Update

Action		Export		
Applications(Components)		Asset.Stock v.1.5 - Asset	Future Trade v.2.5 - Future Trade	Future Trade v.2008 - Future Trade
1	Trade*Net v.6.0.3	RU I	C UDP IO	CRUDP IO
2	SUNGARD TREASURY TRADER v.3	?		
3	salesforce.com v.9			
4	GenLManager v.1.4.6		RU	
5	FX & MM v.3.4			
6	Financial Times v.2.1	CR		

CRUD refers to the four major interactions (Create, Read, Update, and Delete) between architecture elements and business data. Furthermore, the letters I and O (Input and Output) indicate whether the business data are being used in the incoming or outgoing information flows defined for the applications. The notation used to describe business data usage in the CRUD matrix is explained in more detail in the section [Capturing Business Data Usage](#).

The information perspective is used to analyze dependencies between applications due to the data flows between them, especially the dependencies that could impact applications in meeting recovery time objectives, for example. In this case, for example, indicators could be captured that answer the following kinds of questions:

- Which applications process which data?
- Is more than one application creating a particular type of information (CRUD analysis)?
- Which applications have information that must conform to an industry standard?
- Which applications are dependent on others due to information flows feeding them?

Implementing Proposed Information Flows to Manage the Information Architecture

The concept of proposed information flows can be implemented to perform simple changes to the information architecture that don't require the planning and investment needed as in the case of significant architectural changes. For each application or component platform, discovered information flow can be visualized in order to provide up-to-date information about technologies that may be relevant for the technology architecture. Discovered components can be imported on a regular basis from an external discovery system (via CMDBs or a solution such as Dynatrace) and mapped to the relevant application or component in Alfabet. Proposed local components can also be manually created in the context of the application or component they are relevant for. Application/component owners can then assess whether or not a proposed local component shall be included in the platform architecture of the application or component. Proposed local components can be approved and added as "real" local components to the inventory or dismissed and deleted from the Alfabet database. The local components will inherit the values specified for the proposed local components and can be further defined, as needed. Configuration requirements to

implement the proposed local components are described in the section below [Implementing Proposed Local Components to Manage Technology Upgrades and Lifecycles](#).

Chapter 4: Technology Portfolio Governance

The overarching goal of Technology Portfolio Governance is to deliver business value by establishing a catalog of technology standards that are used as fundamental building blocks for developing the enterprise's IT landscape. For example, the application landscape requires a wide range of technologies such as application servers, LDAP repositories, BI engines, etc. These technologies provide the IT infrastructure that the enterprise relies on. But the diversity of these technologies and the constant change introduced by new technologies provides a challenge to maintaining a stable and robust IT infrastructure. Similarly, large and uncontrolled technology portfolios cause complexity and increased costs for maintaining know-how and the help desk.

Technology portfolio governance overcomes this by providing a single repository for technologies and standard technology catalogs that support regional variations. It ensures that standards are adhered to and new technologies are evaluated properly before productive use. As a result, the number of technology combinations to be supported is decreased and thus risks and costs are reduced.

The goals in Technology Portfolio Governance are:

- Govern the component lifecycle and define an approval and decommissioning process for components
- Analyze component lifecycle compliance and usage
- Enforce standards by defining component catalogs and assigning components to mandatory standard platforms
- Reduce technological complexity through reuse and the implementation of standards
- Minimize risk and duration of technology downtime

The following information is available regarding the Technology Portfolio Governance capability:

- [Methodology: Understanding Technology Portfolio Governance](#)
- [Prerequisite: Configuring Evaluation Data for Components](#)
 - [Configuring Evaluation Types](#)
 - [Configuring Aspect Evaluation Types](#)
 - [Configuring Analytics to Understand the Technology Evaluation](#)
- [Importing Components from the Technopedia® Repository](#)
- [Creating, Managing, and Evaluating Component Portfolios](#)
- [Managing Component and Standard Platform Compliance via Component Catalogs](#)
- [Implementing Proposed Local Components to Manage Technology Upgrades and Lifecycles](#)
- [Prerequisite: Configuring the Implementation of Proposed Local Components](#)
- [Managing Device Details and Device Groups](#)
- [Prerequisites: Configuring the Capture of Device Details](#)



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

Methodology: Understanding Technology Portfolio Governance

In general, technology portfolio governance is broken down into the following main tasks:

- **Govern TPG Program:** The governance of the portfolios typically occurs on the corporate level and includes conceptualizing and managing the technology portfolios, ensuring data quality, and establishing the roles and responsibilities for the maintenance of components, component catalogs, and standard platforms, and specifying technology roadmaps.
- **Capture Technology Landscape:** The gathering of information is fundamental to technology portfolio governance. The components in the IT architecture as well as the standard platforms available must be documented. Such components can be conventionally captured, as described in the chapter *Technology Architecture Definition* in the reference manual *Enterprise Architecture Management*, or they can be imported to component categories from the Technopedia® repository. To ensure that the information documented for components is complete, it is recommended that a process is implemented in the enterprise whereby a release status indicates the quality of information about the components and standard platforms. For example, a component with the status "Approved" would indicate to the organization that the component is sufficiently described and is ready for evaluation. An component that is "Approved" could require the following information to be considered complete:
 - Name, description, version
 - Lifecycle and release status definition
 - Business and functional ownership
 - Role assignments that describes the functional relationship or responsibility that a user or organization has to the application.
 - The standard components assigned to component catalogs
 - The standard components assigned to standard platforms
- **Assess Technology Portfolio:** The assessment of the portfolio requires that components are bundled into component groups that logically structure the components to be assessed. The evaluation occurs based on configured indicators used to evaluate the components as well as standard and configured reports. The choice of KPIs and analytics will depend on the goals for the portfolio assessment. For example, if the goal is to reduce technological complexity, then it is necessary to understand the functional quality of components. In this case then, evaluation types could be configured to capture the range of function that is covered by a component, and the quality of the technical services provided. The goals of the portfolio assessment should be carefully considered when conceptualizing and configuring the KPIs to capture for components, component groups, and standard platforms.
- **Plan and Manage Technology Architecture:** The management of the technology focuses on managing the outcomes of the evaluation and, for example, coordinating changes to the component lifecycle, versioning components, and understanding and maintaining the use of components in component catalogs and standard platforms. The tasks of managing component

lifecycles and versioning is described in the context of capturing component data in the chapter *Technology Architecture Definition* in the reference manual *Enterprise Architecture Management*.

Prerequisite: Configuring Evaluation Data for Components

Once the goals of the portfolio assessment have been conceptualized and components have been assigned to the component groups, the details of the evaluation must be specified. This includes the configuration of evaluation types and indicator type. It is useful to consider the various perspectives that are necessary to evaluate the technology landscape and the questions that may need to be answered for each perspective:

- **Business perspective:** The business perspective aims to understand the business usage of components via the application landscape, how critical the components are to the business, and what business requirements will impact the technology portfolio. In this case, for example, indicators could be captured that answer the following kinds of questions:
 - Which components are most/least critical to business operations?
 - Which components provide the most/least value to business?
 - Which components must be agile in order to support changing business needs?
 - Which components are crucial to future business strategies?
- **Technology perspective:** The technology perspective aims to align the technology platform of the component with the enterprise's technology standards and strategy. In this case, for example, indicators could be captured that answer the following kinds of questions:
 - How large is the functional scope of a component?
 - Which components have functional overlap and could be consolidated?
 - Which components have unused functionality?
 - Which components are redundant?
 - Which components are not in line with current technology standards?
 - Which components are easy to maintain?

The following provides an overview of the various mechanisms to evaluate and analyze data in order to assess components in the context of technology portfolio governance:

- [Configuring Evaluation Types](#)
- [Configuring Aspect Evaluation Types](#)
- [Configuring Analytics to Understand the Technology Evaluation](#)

Configuring Evaluation Types

Performance indicators are an important means to understand various aspects of components and standard platforms. In the *Evaluation Page View*, users with access permissions to an object can enter values for customer-defined performance indicators that an authorized user in your enterprise has configured in the

Evaluations and Portfolios functionality. The defined indicators are be used in various standard or configured reports available in Alfabet. Indicators displayed in Alfabet may be either:

- manually entered by a user
- selected from a predefined set of values by a user
- calculated based on other indicators according to computation rules defined by your enterprise
- calculated based on customer-specific code provided by Software AG Support
- imported via an interfacing system such as ADIF

To evaluate and understand the technology portfolio, evaluation types could be configured for the object classes Component, Component Group, and Standard Platform. The indicators that are to be evaluated will depend on the goals of the technology portfolio governance project.

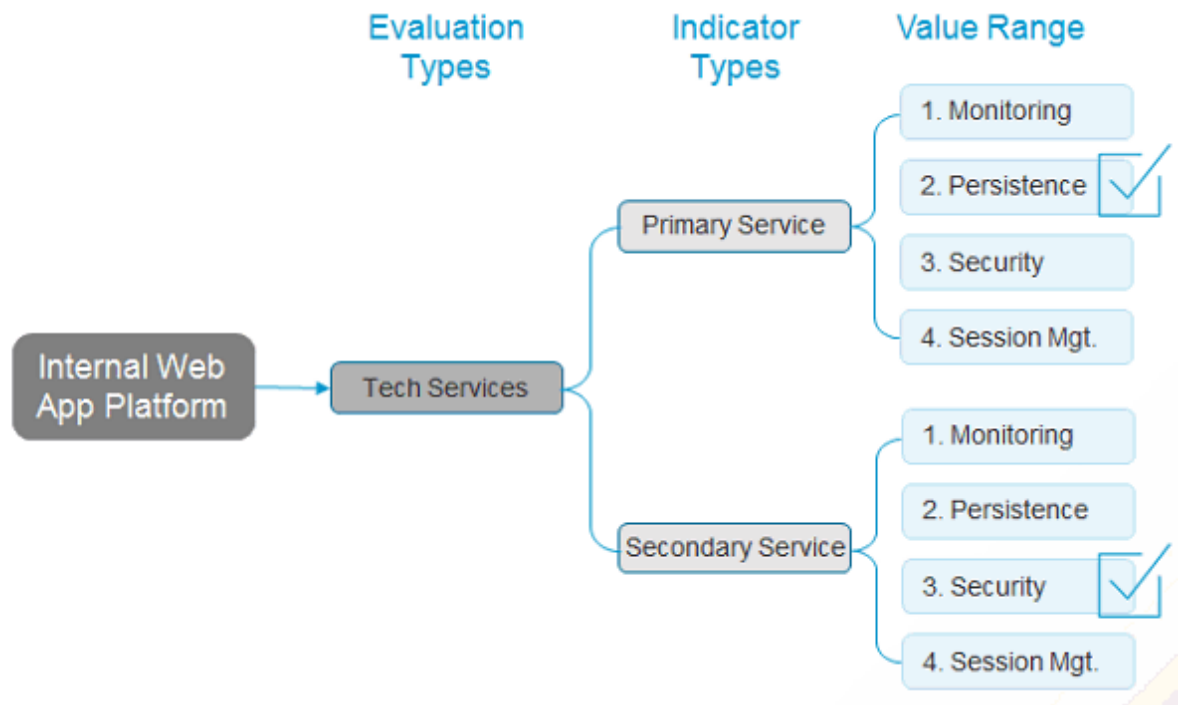


FIGURE: Evaluation of technology support for the standard platform Internal Web App Platform

For example, to understand the technological support that standard platforms provide in the enterprise, an evaluation type Tech Services could be specified with two indicators, one to define the type of support primarily provided and another to provide the type of support secondarily provided by components. In the example above, a value range – a predefined set of values – has been configured that the user must select from to define the indicator. Indicator types having a value range may be associated with an icon gallery, allowing for indicators to be visualized in diagram views as well as other configured reports by means of an icon rather than a numeric value.

Evaluation and indicator types can be configured by a user with access permissions to the **Configuration** module available in the Alfabet interface. Indicator types are bundled in evaluation types and can be reused for a multitude of classes, if necessary. Indicators can also be automatically generated by specifying the calculation (such as sum, minimum, maximum, or average) of multiple indicators or the aggregation of multiple indicators for objects in an object hierarchy. In the case of a computation rule, one or more indicators are bundled in evaluation types that can be aggregated to provide a single value. Depending on your

enterprise's solution configuration, users might manually enter values for each indicator type or select a value from a preconfigured set of values. Some evaluation types may have values that are generated by means of configured computation rules.

For detailed information about how to create and configure evaluation types and indicator types in the **Configuration** module, see the section *Configuring Evaluation Types* in the reference manual *Configuring Evaluation and Reference Data in Alfabet*.

Configuring Aspect Evaluation Types

Another method to evaluate components in Alfabet is to use aspect evaluations. Because a component may be a member of several different component groups, different qualitative assessments of the same component using the same evaluation criteria are possible for each component group that the component is assigned to. Aspect evaluations allow the indicators to be differently defined for components in the context of different component groups.

For example, the component SAP® Solution Manager 4.7 may be relevant for different purposes in the technical architecture. The component may be suitable to varying degrees both as a component of SAP® Business Connector as well as the computing platform SAP Netweaver®. An aspect evaluation would allow you to examine the component from the perspective of the component group SAP BC and the component group SAP Netweaver based on specific aspect evaluation types, such as Functional Quality or Usage Prominence.

Like conventional evaluation types, aspect evaluation type comprise one or more indicator types that are used to evaluate the components. Aspect evaluations can also be analyzed in the context of aspect portfolio reports. For more information about configuring aspect evaluations as well as aspect portfolios, see the section *Configuring Aspect Evaluations for the Classes Application and Component* in the reference manual *Configuring Evaluation and Reference Data in Alfabet*.

Configuring Analytics to Understand the Technology Evaluation

- **Prioritization schemes:** In addition to configuring evaluation types that bundle indicator types, a user with access permissions to the **Configuration** module can also configure prioritization schemes, which bundle one or more sets of evaluation types. A prioritization scheme is a weighted composite of a set of evaluation types and is typically used to determine a prioritized ranking for a set of objects. When the user selects a prioritization scheme in a report, the weighting of the evaluation types is automatically applied to produce a value. A prioritization scheme is typically used to define an axis in a portfolio report.
- **IT portfolios:** A portfolio report is a bubble chart that represents the relative performance of a set of objects that belong to the same object class in two or three independent dimensions of measurement. A portfolio report may display either a conventional X-, Y-, and power-axis or a BCG quadrant. Typical portfolio reports relevant to technology portfolio governance might display a group of components assigned to a component group. In order to work with portfolio reports in Alfabet, IT portfolios must be configured by a user with access permissions to the **Configuration** module. For example, a portfolio shows all components in the component group as bubbles of varying sizes. The X-axis represents the evaluation type Business Important, the Y-axis displays the evaluation type Incidents per Day, and the Z-axis (the bubble size) represents the evaluation type Criticality. The tooltip displays the component's name and version as well as the Z-axis value. The axes of the IT portfolio may represent either a single indicator type, an evaluation type

bundling indicator types, or a prioritization scheme bundling evaluation types. For more information about configuring portfolios, see the section *Configuring Portfolios* in the reference manual *Configuring Evaluation and Reference Data in Alfabet*.

- **Configured reports:** Depending on the configuration of your Alfabet solution, many of the reports that you see may be configured reports that have been created by your solution designer in order to provide a customized data analysis that is not available in the standard Alfabet product. Reports may be configured to display indicators in tables, matrices, portfolios, or Gantt charts. Other graphic visualizations may also be displayed in configured reports including, for example, radar charts, line charts, pie charts, tree map reports, layered diagrams or cluster map reports. For more information about creating configured reports, see the section *Configuring Reports* in the reference manual *Configuring Alfabet with Alfabet Expand*.

Importing Components from the Technopedia® Repository

The gathering of information about the components in the IT infrastructure is fundamental to technology portfolio governance. The enterprise's components can be captured conventionally as described in the chapter *Technology Architecture Definition* in the reference manual *Enterprise Architecture Management*. With the Portfolio Management Basic sales package, however, you can capture components based on products in the Technopedia® repository. To this end, Software AG provides an interface to Technopedia®, a categorized repository of information about enterprise hardware and software.



Alternatively, Technopedia® products can be imported as vendor products rather than components. Your enterprise will determine the methodological approach relevant to importing products from the Technopedia® inventory to Alfabet. Please note that to import products from the Technopedia® inventory as vendor products, you must have access to the Contract and Vendor Management capability which is part of the Portfolio Management Advanced sales package.

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 import Technopedia products to Alfabet. For more information about the configuration requirements as well as the mapping of Technopedia classes to Alfabet classes, see the section *Configuring Interoperability with Technopedia* in the reference manual *Configuring Alfabet with Alfabet Expand*.

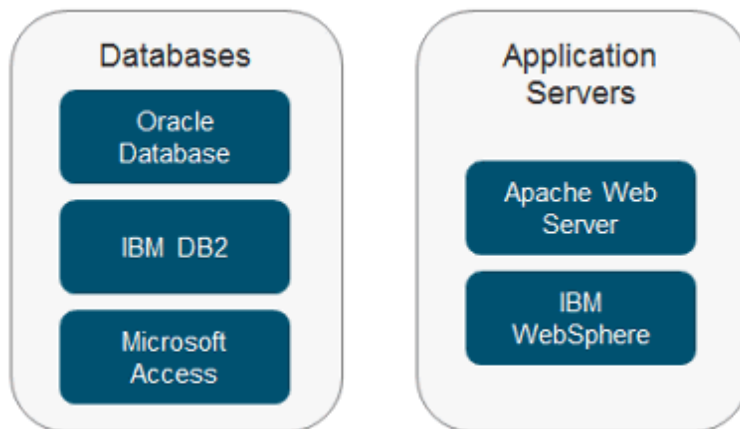
By importing Technopedia® products to Alfabet, components can be captured in a standardized manner and the technology information will be aligned across the enterprise. The catalog of structured information about the IT infrastructure ensures a unified language and discipline to manage and plan the technology 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.

The products in the Technopedia® repository are structured in product categories. When you import product categories from Technopedia®, you import the entire repository of Technopedia® product categories. The Technopedia® products can then be imported on a category-by-category basis, as needed. The imported product categories and products will be added respectively as component categories and components. Each component category will be associated with a Technopedia ID. Any component categories that have already been defined in Alfabet will remain in the Alfabet database and continue to be available for definition in the **Components** explorer. While it is technically possible to create component categories in Alfabet and implement categories from Technopedia®, it is recommended that your enterprise use only one source of component categories. You may however, import Technopedia products to any component category that has been created in Alfabet.

Please note that in the Technopedia® repository, components are assigned only to the leaf-level component categories. The product category hierarchy in Technopedia® consists of two levels. The top level of the Technopedia® product categories will be imported to the top level of the **Components** explorer in the Alfabet interface and the subordinate product category will be nested below. Component categories that have been imported via Technopedia® will initially have no components assigned to them. In other words, you must explicitly create the components based on the Technopedia products for each relevant component category, as needed. To add Technopedia® products to component categories that have been imported to Alfabet, you must navigate to the object profile/object cockpit of the leaf-level component category and add the components to it via the *Components Page View*.

Creating, Managing, and Evaluating Component Portfolios

Component portfolios are defined in Alfabet via component groups that bundle components. The components should be bundled in component groups that are relevant to the goals for the portfolio assessment process. A component can be assigned to multiple component groups.



For example, components could be logically structured in component groups according to the technologies that the components support. For example, application servers or the databases actively in use in the enterprise could be grouped in order to assess them to find out which might be shut-down candidates in the near future.

Component groups are created at the root node of the *Component* explorer. Subordinate groups are created in the *Sub-Groups Page View* of the parent component group. A number of governance concepts are implemented for component groups:

- **Authorized User:** Each component group has an authorized user. An authorized user has primary responsibility for the component group 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 an component group will have Read/Write access permissions to the component.
- **Roles:** A role defines the functional relationship or responsibility that a user or organization has to a component group (for example, the Operator or Owner). Various roles may be required in order to provide input to the technology portfolio from various perspectives. Roles describe responsibilities but they do not authorize access permissions to the component group in Alfabet.

- **Mandates:** Component groups may be managed in a federated architecture. By means of mandates, it is possible to specify the visibility of individual component groups in the Alfabet interface for specific users. The mandate capability may be activated or deactivated for the entire capability as well as explicitly activated or deactivated for specific component group stereotypes.
- **Object Class Stereotypes:** Object class stereotypes may be configured by your solution designer for the object class **Component Group**. This allows for a different governance approach between different kinds of component groups. If object class stereotypes are configured for the object class **Component Group**, each stereotype may capture a specified set of attributes, reference data, and class configurations as well as implement a different governance approach.



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*.

Once components have been assigned to component groups in the *Components Page View* of the relevant component group and evaluation types and indicator types have been configured for the object class **Component**, the actual evaluation process can begin. Typically, each component will be evaluated by its authorized user or a member of the authorized user group in the *Evaluation Page View* in the component's object profile. The view will list all evaluation types and their indicator types that have been assigned to the object class **Component**. Depending on the configuration of the indicator type, the user may be required to enter a numerical indicator or select a value in the drop-down list. It may be required by your enterprise that you explicitly define indicators for a component group as well. The *Evaluation Page View* is also available in the object profile of a component group and allows component groups as a whole to be evaluated.

Once the components have been evaluated, you can look at various reports that allow you to analyze the component portfolio from a variety of perspectives. For example:

- The *Components Portfolio Page View* allows you to analyze all components in a selected component portfolio based on configured IT portfolios and the *Sub-Group Portfolio Report* allows you to understand the assessment of all component groups subordinate to a selected component group.
- The *Aspect Portfolio Page View* allows you to examine qualitative information about the components in the selected component group based on various configured aspects.

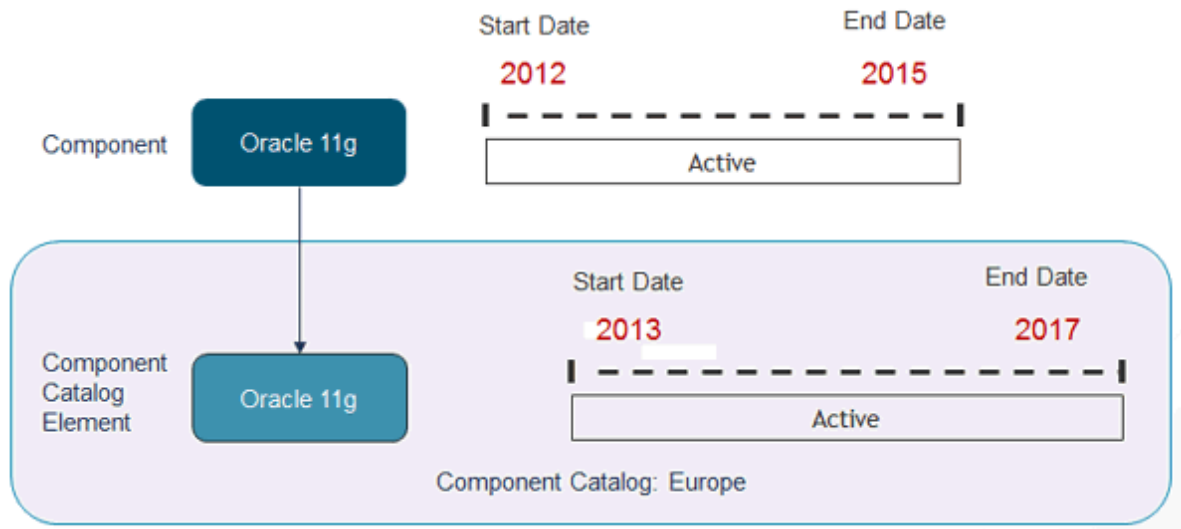


In addition to the standard reports listed above, your enterprise may have configured reports that are specifically geared to the assessment and reporting needs of your enterprise. Your enterprise must purchase the relevant license to the expansion set **Custom Reports**. For more information about configuring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

Managing Component and Standard Platform Compliance via Component Catalogs

A catalog of technology components and standard platforms allows IT standards to be communicated and managed across the enterprise. In Alfabet, component catalogs can be created to maintain components and standard platforms from the perspective of a geographical region, business unit, or product portfolio, for example. The component can thus belong to more than one component catalog and have different lifecycles in different component catalogs. Any component assigned to a component catalog is considered

a component catalog element. Every component catalog element has a lifecycle independent of its associated component.



In the context of a component catalog, you can understand the level of standardization and compliance of components and standard platforms for a defined period of time. For example, an enterprise has purchased the vendor product Oracle® Database with the start and end dates of 2007–2019. The active period of the component Oracle 11g is defined as 2012–2015. However, a business unit OptiRetail Strategy, Marketing & Sales uses the application OptiRetail Marketing Solution. This application requires the component Oracle 11g for the application platform which is implemented from 2013 until 2017. In this example, the component Oracle 11g can be assigned to a component catalog created for the business unit OptiRetail Strategy, Marketing & Sales and be assigned the start and end dates, which are independent of the base components start and end dates. In the context of the component catalog OptiRetail Strategy, Marketing & Sales, Oracle 11g is a component catalog element.

Component catalogs are created in the *Component Catalogs Explorer*. A number of governance concepts are implemented for component catalogs:

- **Authorized User:** Each component catalog has an authorized user. An authorized user has primary responsibility for the component catalog 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 component catalog will have Read/Write access permissions to the ICT object group.
- **Roles:** A role defines the functional relationship or responsibility that a user or organization has to a component catalog (for example, a Program Manager or Budget Owner). Various roles may be required in order to provide input to the technology portfolio from various perspectives. Roles describe responsibilities but they do not authorize access permissions to the component catalog in Alfabet.
- **Mandates:** Component catalogs may be managed in a federated architecture. By means of mandates, it is possible to specify the visibility of individual component catalogs in the Alfabet interface for specific users. Mandates are typically configured if component catalog stereotypes are configured. The mandate capability may be activated or deactivated for the entire capability as well as explicitly activated or deactivated for specific ICT object group stereotypes.



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*.

Components and standard platforms can be added to a component catalog in the *Components Page View* and the *Standard Platforms Page View* respectively. Each component is considered a component catalog element and each standard platform is considered a standard platform catalog element in the context of the component catalog. You will be required to explicitly define the start and end dates of each component catalog element or standard platform catalog element for the selected component catalog. In this way, you can add the same component/standard platform to multiple component catalogs and each component catalog element/standard platform catalog element may have a different lifecycle definition. The lifecycles of all component catalog elements and standard platform catalog elements assigned to a selected component catalog can be reviewed in the *Component Catalog Elements Lifecycle Page View*.

Once components and standard platforms have been assigned to component catalogs, you can then assess whether, for example, the components in an application's platform are compliant with the relevant technology roadmap in the *Platform Architecture Page View*. The technology roadmap could be a corporate roadmap as documented in the list of components managed in Alfabet, or a roadmap defined for a specific entity, purpose, or region as documented in a component catalog. The compliance of the platform elements is always validated with regard to a specified date (current or future).

Application: OptiRetail Marketing Solution 2.1

Platform Architecture

Component Catalog: Europe Lifecycle Status: Core Date: 30/12/2017 Update

Navigation: Navigate View Move Export

	Client Tier	Presentation Tier	Business Tier
Business Layer	Adobe Acrobat Reader SAP GUI v.4.7 c Oracle Database v.11g		SAP CO v.4.7 c SAP FI v.4.7 c SAP MM v.4.7 c SAP PM v.4.7 c
Software IS	Microsoft Windows v. CORBA Microsoft Internet Ex Progress ORBIX	Linux CORBA Apache Apache Web Server	WebSphere Linux CORBA (J2EE) IBM Websphere

FIGURE: Oracle 11g is compliant with the selected component catalog Europe

The report above displays that the component Oracle 11g is compliant on the selected date and for the lifecycle state Core according to its definition in the component catalog Europe. In the *Platform Components Lifecycle Page View*, you can review the compliance of the lifecycle of all platform components assigned to the selected application according to a selected component catalog. In this way, the enterprise can determine if a component is approved as a standard component, has limited support, or is not permitted for future use, etc.

Implementing Proposed Local Components to Manage Technology Upgrades and Lifecycles

The concept of proposed local components can be implemented to perform simple technology changes that don't require extensive planning and investment. Local components discovered from other systems (such as Dynatrace® or AppDynamics®) and imported to Alfabet may be created as proposed local components and only added as real local components to the organization's IT inventory after they have been explicitly included in an approval process. Proposed local components can also be manually created in the context of the application or component they are relevant for. Application/component owners can then assess whether or not a proposed local component shall be included in the platform architecture of the application or component. Proposed local components can be approved and added as "real" local components to the inventory or dismissed and deleted from the Alfabet database. The local components will inherit the values specified for the proposed local components and can be further defined, as needed. Configuration requirements to implement the proposed local components are described in the section below [Implementing Proposed Local Components to Manage Technology Upgrades and Lifecycles](#)

The following scenarios are relevant to the implementation of proposed local components:

- Create integration solutions that glean local components from operational systems. Local components can be created as proposed local components upon import. Users can approve and accept them thereby turning them into real local components.
- Introduce proposed local components in the architecture platforms of applications and components without having to create newer versions of the owning applications/components. For example, for an upgrade from SQL Server 2016 to SQL Server 2019, a user could simply introduce a proposed local component in the same application platform (and possibly with start and end date that is different from the application or predecessor local component (rather than creating a completely new application).

If your solution is configured to implement the proposed local component concept, the proposed local components will be displayed in the *Components Page View* as well as the *Platform Architecture Page View* of the relevant application or component. The following decisions can be made about the inclusion of the proposed local components:

- You can specify that a proposed local component shall be included in the platform whereby it is added as a real local component to the Alfabet database.
- You can specify that a proposed local component can be dismissed whereby it is deleted from the Alfabet database.
- You can specify that a proposed local component can be selected to replace an existing local component whereby the proposed local component becomes a real local component and the selected component is deleted from the Alfabet database.

The *Technical Platform Elements Page* will display proposed local components along with all other platform elements that have been added to the platform.

Prerequisite: Configuring the Implementation of Proposed Local Components

The following configuration is required to implement the proposed local component concept:

- In order to implement the concept of proposed local components (and proposed information flows) in your solution configuration, the XML attribute `EnableProposedObjectHandling` must be set to "true" in the XML object ***SolutionOptions*** available in the configuration tool Alfabet Expand. This setting ensures that imported and manually created proposed local components as well as proposed information flows will be visualized in the relevant views and the menu options to create and edit them will be available in the Alfabet user interface. For more information about this configuration, see the section *Implementing Proposed Local Components and Proposed Information Flows* in the reference manual *Configuring Alfabet with Alfabet Expand*.
- Proposed local components (as well as proposal information flows) may be imported from discovery tools (such as Dynatrace® or AppDynamics®) via an ADIF import scheme but can also be manually added by a user. In this case, the components discovered in the external system must be imported as objects of the class `ProposedLocalComponent` and mapped to the applications or components for which they are relevant. For more information about configuring an ADIF import scheme to import discovered local components and information flows from another system, see the reference manual *Alfabet Data Integration Framework*.

Managing Device Details and Device Groups

In some contexts such as PaaS and IaaS, the device infrastructure may need to be captured in more granular detail. The notion of devices using other devices and being used by other devices can be captured as a device composition. A device composition describes other devices such as storage, routing, or queuing service devices that are relevant for a current device. Typically, object class stereotypes will be configured for the class **Device Composition** to capture various device compositions. The composition of a device can be documented in the *Used Devices Page View* that provides a top-down approach where you can capture the devices that a selected device uses, and the *Using Devices Page View* that provides a bottom-up approach where you can specify the devices that are using a selected device.

Further detail can be captured for devices via the *Device Details Page View*. A device detail captures detailed information for devices such as volumes, DNS entries, or network cards. Typically object class stereotypes will be configured for the class **Device Detail** to provide relevant attributes to capture different information.

Furthermore, devices can be bundled in device groups that are relevant to the goals for the portfolio assessment process. A device can be assigned to multiple device groups. For example, devices could be logically structured in device groups according to the services the device support such as elastic cloud devices or edge devices. Device groups are created at the root node of the *Devices by Group Explorer* explorer. Subordinate groups are created in the *Device Groups Page View* of the parent device group.

A number of governance concepts are implemented for device groups:

- **Authorized User:** Each device group has an authorized user. An authorized user has primary responsibility for the device group 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 device group will have Read/Write access permissions to the device.
- **Roles:** A role defines the functional relationship or responsibility that a user or organization has to a device group. Various roles may be required in order to provide input to the technology portfolio from various perspectives. Roles describe responsibilities but they do not authorize access permissions to the device group in Alfabet.

- **Mandates:** Device groups may be managed in a federated architecture. By means of mandates, it is possible to specify the visibility of individual device groups in the Alfabet interface for specific users. The mandate capability may be activated or deactivated for the entire capability as well as explicitly activated or deactivated for specific device group stereotypes.
- **Object Class Stereotypes:** Object class stereotypes may be configured by your solution designer for the object class **Device Group**. This allows for a different governance approach between different kinds of device groups. If object class stereotypes are configured for the object class **Device Group**, each stereotype may capture a specified set of attributes, reference data, and class configurations as well as implement a different governance approach.

Prerequisites: Configuring the Capture of Device Details

The following configuration is possible to capture granular information about devices:

- Object class stereotypes should be configured for the class **Device Composition**. A device composition describes other devices such as storage, routing, or queuing service devices that are relevant for a current device.
- Object class stereotypes should be configured for the class **Device Detail**. A device detail captures detailed information for devices such as volumes, DNS entries, or network cards. The configuration of custom properties and custom editors allow the relevant information to be captured for each device detail stereotype.
- Object class stereotypes may be configured for the class **Device Group**. A device group is a container to logically structure devices in order to view, analyze, and communicate the company's physical IT infrastructure.

Chapter 5: Business Capability Management

The Business Capability Management capability allows you to capture and specify the functional domains in your enterprise in order to identify and understand how the functional architecture supports your enterprise's business goals. By documenting the enterprise's functional domains, you can attain transparency about which aspects of the IT architecture including, for example, applications, components, and business processes provide functionality to the enterprise as well as pinpoint areas of functional inefficiency and redundancy in the IT architecture.

Some enterprise's capture their functional architecture by means of business capabilities, which 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. Please note that if your enterprise captures its business by means of business capabilities, you must first capture these business capabilities by means of the object class Domain in Alfabet. The domains/business capabilities you capture can then be evaluated by relevant members of your enterprise and analyzed in the context of a business capability map. Alfabet provides a number of standard reports that allow you to analyze your enterprise's business capabilities in order to understand the applications, technologies and projects that support the business so that IT changes won't adversely impact business operations.

The following information is available:

- [Documenting the Enterprise's Functional Domains](#)
 - [Methodology: Understanding Domain Models](#)
 - [Prerequisite: Configuring the Domain Management Capability](#)
 - [Configuration of Domain Stereotypes](#)
 - [Configuration of Evaluation Data](#)
 - [Configuration of Frameworks \(Optional\)](#)
 - [Understanding Governance and Responsibility for Domain Models](#)
 - [Documenting and Defining the Domain Model](#)
 - [Assigning Architecture Objects to a Domain](#)
 - [Defining Business Functions and Business Services](#)
 - [Defining the Operational Business Support for a Domain](#)
 - [Documenting the Service Architecture for a Business Process](#)
 - [Analyzing the Domain Model](#)
- [Documenting the Enterprise's Business Capabilities](#)
 - [Prerequisite: Configuring the Business Capability Management Capability](#)
 - [Configuration of Domain Stereotypes](#)
 - [Configuration of Evaluation Data](#)
 - [Defining Business Capabilities and Creating a Business Capability Map](#)
 - [Evaluating the Business Capabilities](#)

- [Analyzing the Enterprise's Business Capabilities](#)
- [Capturing Business Appraisals in the Enterprise](#)

Documenting the Enterprise's Functional Domains

In order to understand the functions that IT provides to the business and thus the value of the functional architecture to the enterprise's business goals, you need to identify and describe the various functional domains of the business. In the broad sense, functional domains describe the functional blocks in the architecture that support the enterprise to carry out its business goals. These blocks of functionality are captured as "domains" in Alfabet.

Domains allow the enterprise architecture to be hierarchically partitioned into disjoint segments. An enterprise may have multiple domain models that bundle the domains from a particular perspective, as shown in the figure below.

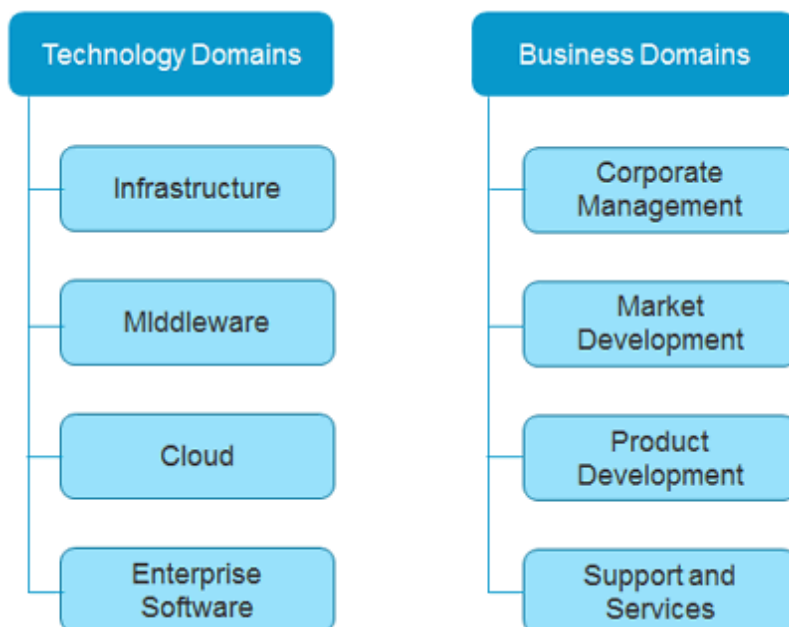


FIGURE: Two domain models in an enterprise

For example, an enterprise may have a domain model Technology Domains that captures the technological functionalities that support the enterprise and a domain model Business Domains that documents the functional blocks of the IT that are relevant to support the enterprise's areas of business that support the enterprise.

The following information provides you with an overview of the concepts relevant to structuring your enterprise's domain models as well as an overview of the configuration required. Information is also provided about how to capture and define the basic information necessary to analyze the domains in your enterprise.

The following information is available:

- [Methodology: Understanding Domain Models](#)
- [Prerequisite: Configuring the Domain Management Capability](#)

- [Configuration of Domain Stereotypes](#)
- [Configuration of Evaluation Data](#)
- [Configuration of Frameworks \(Optional\)](#)
- [Understanding Governance and Responsibility for Domain Models](#)
- [Documenting and Defining the Domain Model](#)
- [Assigning Architecture Objects to a Domain](#)
- [Defining Business Functions and Business Services](#)
- [Defining the Operational Business Support for a Domain](#)
- [Documenting the Service Architecture for a Business Process](#)
- [Analyzing the Domain Model](#)



Please note that a context-sensitive Help is available for each view available in the Business Capability 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 Domain Models

The domain model describes a hierarchy of domains in the enterprise. Each domain represents a functional entity in the domain model and allows the enterprise architecture to be hierarchically partitioned into disjoint segments and structured from a specific point of view. An enterprise may have one or more domain models, each with a distinct hierarchy of domain stereotypes.

In the configuration tool Alfabet Expand, your solution designer will configure the object class stereotypes that are relevant for your enterprise. When a user captures a domain, he/she will be required to identify the stereotype that it is based on. Each object class stereotype may be configured to capture a different set of attributes for the domains based on that stereotype as well as provide a different set of relevant views and reports.

Typically, each level in the domain hierarchy will represent a different domain stereotype. In the example below, three stereotypes have been configured for the domain hierarchy: These three stereotypes have been specified with the names Area, Sub-Area, and Domain.



Please note that your solution designer will specify the naming of the domain stereotypes. The documentation, however, refers to all domains based on any domain stereotype using the generic term "domain".

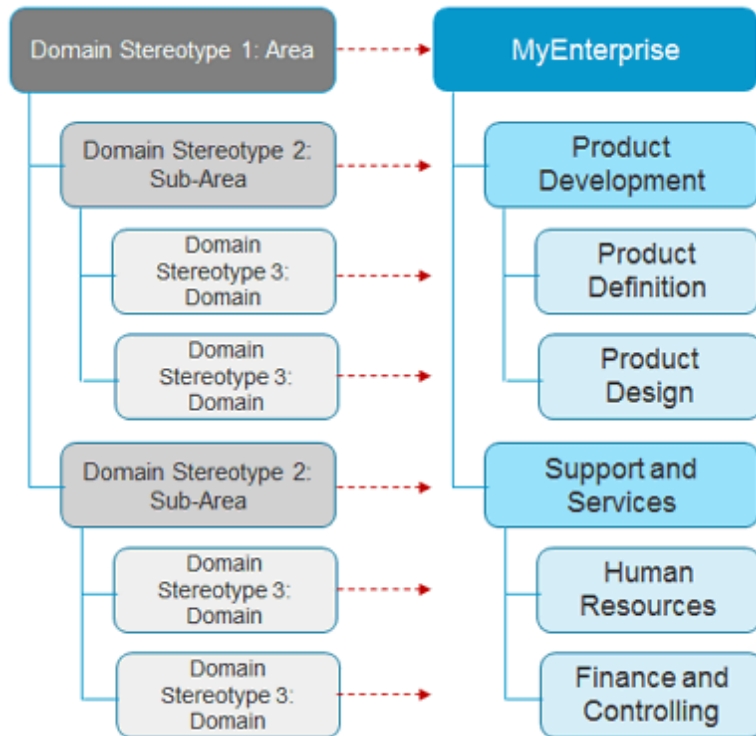


FIGURE: Domain stereotypes in domain hierarchy

The image in grey represents the configured hierarchy of domain stereotypes and the image in blue represents an example of an actual domain hierarchy as it might be defined in the Alfabet interface.

Depending on the domain stereotype, a domain may own business functions, business objects, business processes, applications, ICT objects, components, standard platforms, and vendor products. Your solution designer will determine which object classes may be associated with the domain stereotype. One goal to achieve an effective service-oriented architecture is to structure the IT landscape so that each architecture element is owned by only one domain, which defines the principal functional context for the object. In this case, the architecture element is considered a primary object of the domain.

For example, the solution designer may configure the domain stereotype Domain to allow applications and business functions to be assigned to it.

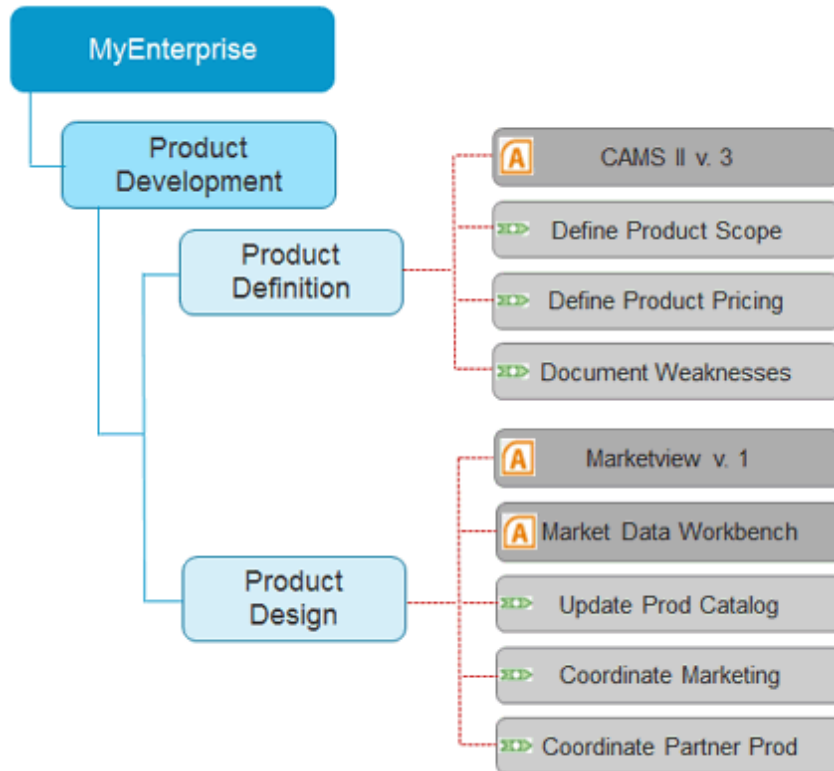
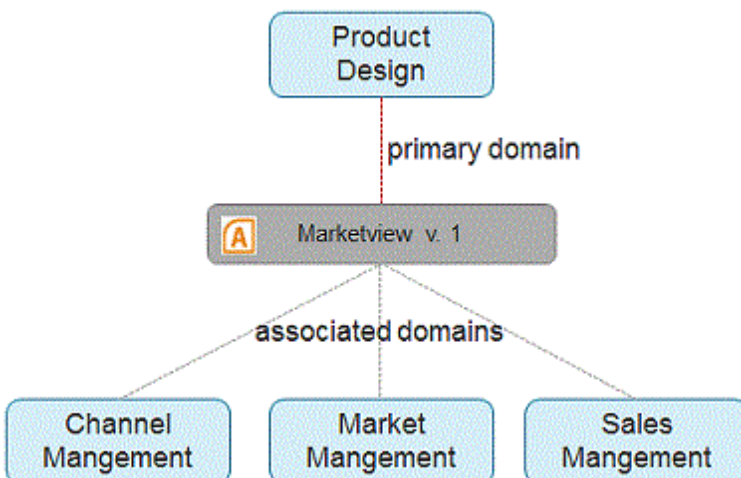


FIGURE: Applications and business functions assigned to domains

In the figure above, the domain Product Definition is the owner of the application CAMS II v.3 and the business functions Define Product Scope, Define Product Pricing, and Document Product Weaknesses.

Although each architecture element can only be owned by one primary domain, in a real-life enterprise, an architecture element may actually have a functional role in multiple domains. To capture this context, an architecture element may also be considered an associated object of other domains. An associated object of a domain is an architecture element that is assigned to a domain without being owned by that domain. Depending on the solution configuration, business functions, business objects, business processes, applications, ICT objects, components, standard platforms, market products, or vendor products may be assigned as associated objects to multiple domains of a specified domain stereotype.



For example, the application Marketview v. 1 is owned by the domain Product Design but may support several other business activities. In this case, the application Marketview v. 1 could be defined as an associated

object of the domains Channel Management, Market Management, and Sales Management. Please note that it is recommended that an object has a minimal number of associated domains. Multiple associated domains indicate that the IT landscape should be optimized for improved reusability.



A **Domain Planning** functionality allows you to plan and design an alternative hierarchy of domains for a section of the domain model in the context of a solution domain project. In the solution domain project, you can restructure the domain hierarchy by creating new domains, deleting existing domains, and proposing successor domains for the objects referenced by the deleted domains without actually modifying the currently implemented domain model. When the design of a solution domain project has been approved for the enterprise, it can be checked back in to the Alfabet inventory, thus overwriting the corresponding section of the domain model. To work with solution domains, you must have access to the Scenario Management capability which is part of the IT Planning Complete sales package.

Prerequisite: Configuring the Domain Management Capability

In order to work with the domain management capability, the following configuration is required:

- [Configuration of Domain Stereotypes](#)
- [Configuration of Evaluation Data](#)
- [Configuration of Frameworks \(Optional\)](#)

Configuration of Domain Stereotypes

The domain model capability is highly configurable and can be fitted to the individual specifications of your enterprise. Via the configuration tool Alfabet Expand, your enterprise's solution designer can create and configure one or more domain models, each with different domain stereotypes. For detailed information about the necessary configuration that is carried out in the configuration tool Alfabet Expand, see the section *Configuring Domain Models and Domain Planning* in the reference manual *Configuring Alfabet with Alfabet Expand*.

Please be aware that the following configuration issues will impact the definition of domain models in your enterprise:

- Your solution designer may configure multiple domain models and each domain model may have a different number of levels in the domain hierarchy. Each domain model may consist of different domains based on different domain stereotypes. For example, your enterprise may consist of two domains models, a Technology Domain with two hierarchy levels (two domain stereotypes) and a Business Domain with three hierarchy levels (three domain stereotypes).
- Your solution designer will configure object class stereotype for the class **Domain** for each hierarchical level in your enterprise's domain model. Each object class stereotype may have only one ascendant stereotype but may have multiple subordinate stereotypes. The name, number, and hierarchical ordering of the stereotypes are configurable. Each domain that a user creates in Alfabet must be based on a domain stereotype.
- Your solution designer may configure a domain stereotype to be recursive, thereby allowing subordinate domains to be defined that are based on the same domain stereotype.

- Your solution designer may configure which architecture elements may be displayed in the **Domains** explorer as well as which be created for or assigned to a domain. The permissible architecture elements may differ for each domain stereotype. These configurations will determine which architecture elements can be defined as primary objects of a domain and which can be defined as associated objects of a domain.
- Whereas the documentation provided by Software AG uses the generic term "domain" for all domains regardless of the domain stereotype they are based on, your enterprise may use different terminology for each level in the domain model. For example, the first level may be named Areas, second level Sub-Areas, third level Domains, etc.

Configuration of Evaluation Data

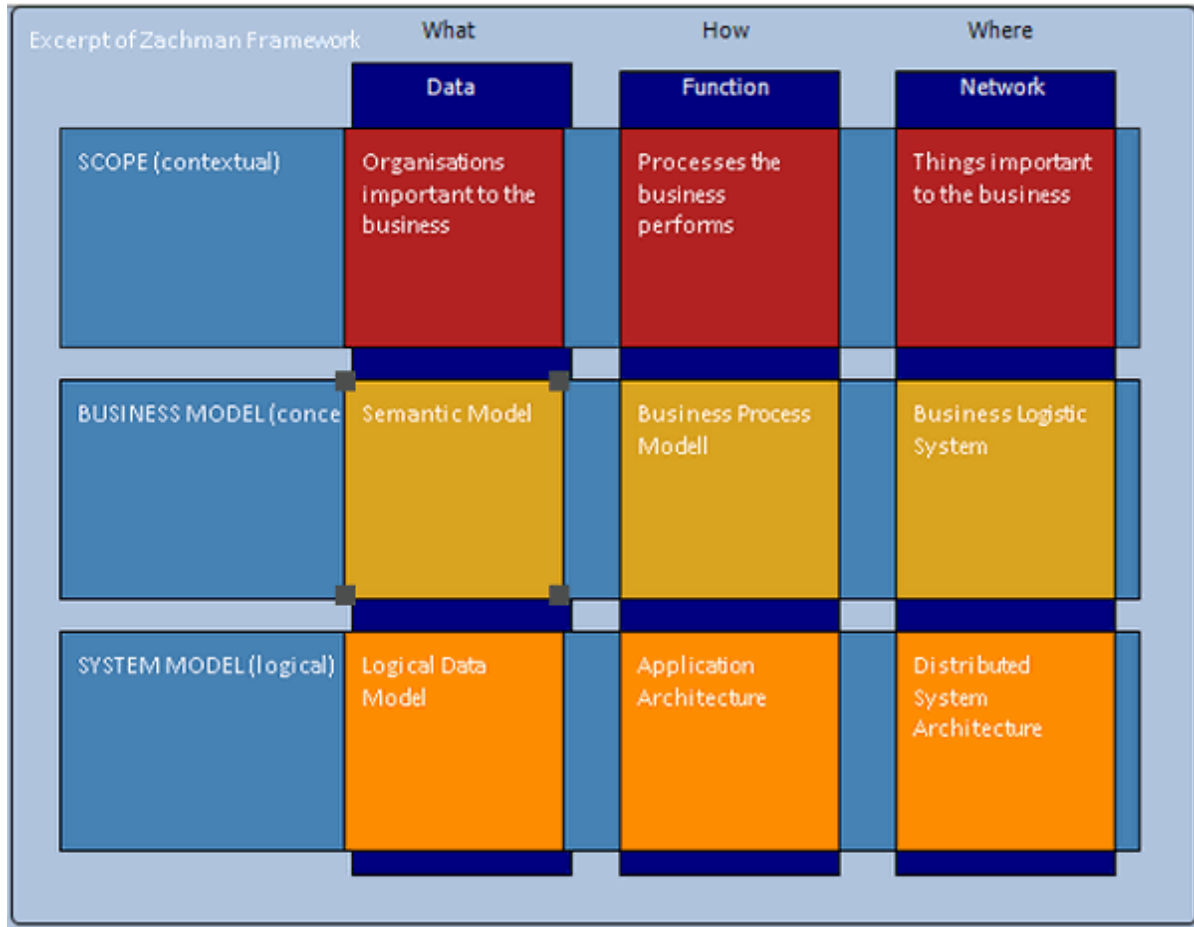
In addition to configuring domain stereotypes in the configuration tool Alfabet Expand, a user in your enterprise with access permissions to the **Configuration** module will typically configure evaluation and reference data that will allow the user community to evaluate and analyze the functional landscape. Your enterprise can determine the KPIs relevant to assess the functional architecture.

The evaluation types that are to be configured for the object class Domain will depend on your enterprise's assessment needs. For example, you could define an evaluation type that assesses the operational costs of a domain. Or to understand the current and planned support of the functional landscape, you might want to understand the current and planned operational support provided by applications to the domains. In this case, you could configure an evaluation type Domain Statistics that bundles the indicators Assigned Applications (Current), which indicates the number of applications currently owned by the domain, Assigned Applications (Planned), which indicates the number of applications that are planned for the domain, and Business Support Applications, which indicates the number of applications that provide support to the domain.

The indicators can then be displayed in various standard or configured reports. IT portfolios can be configured to display objects in a bubble chart based on their indicators. For example, a portfolio could be configured to assess the indicators Market Differentiation, Mission Criticality and Operational Costs in order to understand the business significance of the functional domains in the enterprise architecture. For detailed information about how to create and configure evaluation types, indicator types, and IT portfolios, see the chapter *Configuring Evaluations, Prioritization Schemes, and Portfolios* in the reference manual *Configuring Evaluation and Reference Data in Alfabet*.

Configuration of Frameworks (Optional)

Frameworks may be designed by your enterprise to provide access to data, and model and present objects in a specific structure that may be different from the structure of the inventory objects. Many companies use this feature to configure a standard framework like Zachmann or TOGAF® or a reference model that is domain specific.



The framework allows you to represent a semantic structure for objects managed in Alfabet, regardless of the object class that an object belongs to. This form of presentation might be particularly useful to visualize a domain-specific reference model that allows users to navigate through the various levels of the functional architecture. Users can double-click a cell representing a domain, for example, to navigate to more detailed information in Alfabet, thus moving through the domain hierarchy to access the architecture elements that the domain owns or is associated with. Frameworks are typically viewed by a large number of users that may have no direct access to the inventory of objects in Alfabet.

The framework is used solely for the purposes of visualization, presentation, and navigation and constitutes a hierarchically-structured collection of framework groups designed in the context of a framework diagram. A framework group structures objects from different object classes and may have subordinate framework groups that also contain objects. The framework diagrams and reference models are available to users in the **Corporate Frameworks** functionality. For more information about configuring and working with framework diagrams and reference models, see the section *Defining and Navigating with Your Corporate Frameworks* in the reference manual *Getting Started with Alfabet*.

Understanding Governance and Responsibility for Domain Models

A number of governance concepts are implemented for domains.

- **Authorized User:** Each domain has an authorized user. An authorized user has primary responsibility for the domain 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 domain will have Read/Write access permissions to the domain.

- **Mandates:** Domains may be managed in a federated architecture. By means of mandates, it is possible to specify the visibility of individual domains in the Alfabet interface for specific users.
- **Object Class Stereotypes:** Object class stereotypes may be configured by your solution designer for the object class Domain. This allows for a different governance approach between different kinds of domain such as, for example, Business Domains and Technology Domains. If object class stereotypes are configured for the object class Domain, 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 domain. In the context of domains, for example, Enterprise Architects, Domain Architects, Project Managers, and Demand Managers may be required in order to provide relevant input into the functional domains in the enterprise. Roles describe responsibilities but they do not authorize access permissions to the domain in Alfabet.



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*.

Documenting and Defining the Domain Model

The domain model constitutes the entire hierarchy of domains, starting with the root domain. Domains at the top level of the domain hierarchy are captured via the root node of the *Domains Explorer*. These domains should be described broadly. The domains become more detailed on the level of the subordinate domains. Each subordinate domain is created in the *<Sub-Domains> Page View* of its parent domain. The domain is created and defined by means of the **Domain** editor.

The screenshot shows the 'Domain editor' interface with the 'Basic Data' tab selected. It contains the following fields:

- ID:** DOM-30
- Level ID:** A.5
- Name*:** Support and Services
- Description:** Infrastructure and other basic services to ensure the lights are kept on

FIGURE: Domain editor to define the domain Support and Services

- It is recommended that each domain has a unique name.
- Each domain requires a number describing its level in the domain hierarchy.
- You should provide a description of the domain so that other users understand its purpose.
- As the creator of the domain, you are automatically defined as the authorized user per default. The authorized user of the domain can be changed in the **Authorized Access** tab. You can also define any user groups that should have Read/Write access permissions to the domain in the **Authorized Access** tab.

Assigning Architecture Objects to a Domain

One of the goals to achieve an effective service-oriented architecture is an IT landscape whereby each architecture element is assigned to only one domain. Ideally, each ICT object, application, component, business function, business process, business object, standard platform, and vendor product should be assigned to one primary domain. In this case, the primary domain owns the architecture element. Your solution designer will typically specify which architecture elements (or their object class stereotypes) may be owned by which domain stereotype.

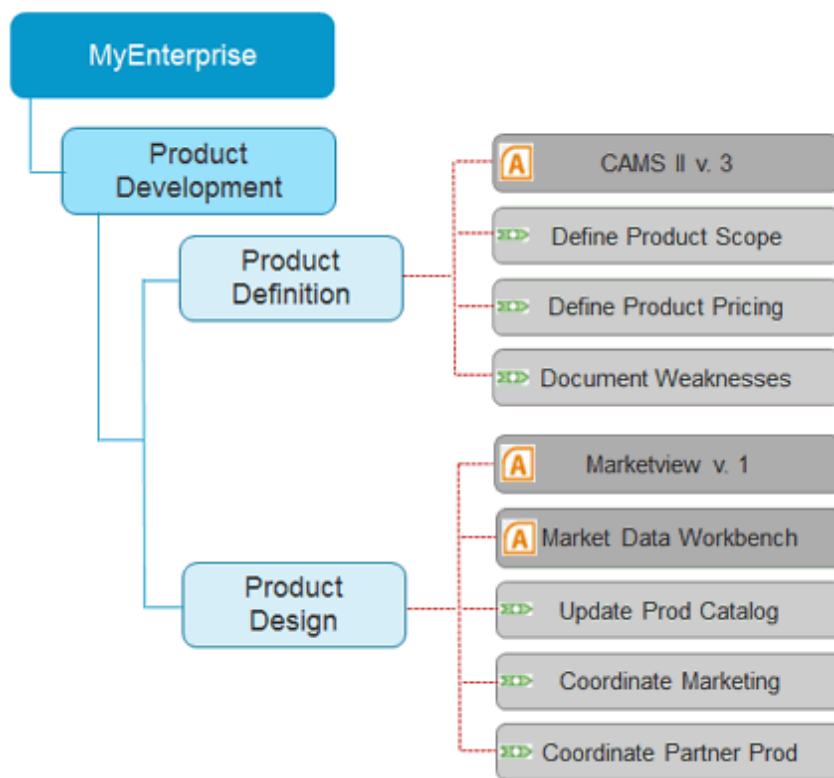


FIGURE: Applications and business functions assigned to a primary domain

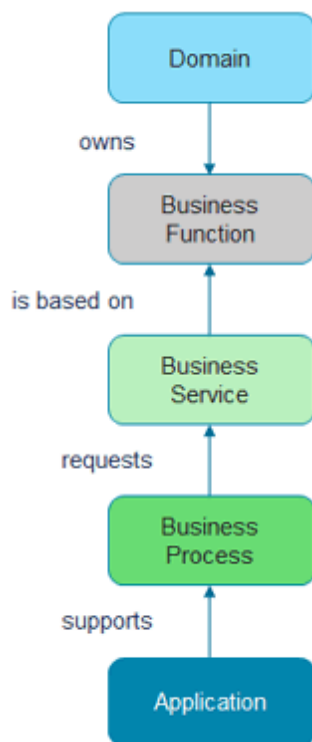
In the example above, the domain Product Definition owns the application CAMS II as well as the business functions Define Product Scope, Define Product Pricing, and Document Product Weaknesses. The domain can be specified as the primary domain of the application in either the **Application** editor or in the *Applications Page View* page view available in the domain's object profile.

Although a clearly delineated service-oriented architecture is the ideal, in reality an architecture element such as an application, ICT object, component, standard platform, etc. is often relevant for multiple

domains. In Alfabet, such an architecture element can also be specified as an associated object of one or more domains. For example, an application may be used in different functional contexts and thus associated with several different non-primary domains. In this case, the application is considered an associated object of those domains. It is recommended that your enterprise keep the number of associated domains for each object to a minimum. Multiple associated domains indicate that the IT landscape can be optimized for improved re-usability.

Defining Business Functions and Business Services

A business function is typically owned by a domain and describes the functional structure of the enterprise architecture.



A business function denotes an atomic business activity that is typically performed in the scope of one or more business processes. For example, the business functions Order Validation, Securities Transaction Processing, and Trade Risk Analytics might be performed for the business processes Hedging as well as Deal Structuring. A business function is characterized by its independence from the business process context and may be applicable across different business processes. A business process may reference one or more business functions through respective business service requests and can thus be understood as the lowest-level building blocks for constructing business processes. It may be that several applications support the same business functions, but it is also possible that a set of applications together are necessary to technically support a business function. Business functions are defined in the *Business Functions Page View* of a domain or business function category, which structure business functions for purposes of organization and analysis.

The business function concept is fundamental to implementing a service-oriented architecture. A business service is an IT service that can be provided by an application, component, local component, organization, market product, business process, ICT object, or solution building block in order to realize a specific business function. The business service fulfills a specific business service request that is posed by a business

process for that specific business function. Thus, 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. Business services are defined in the *Business Services Page View* of the application or component providing the business service or the *Business Service Requests Page View* of the business process requesting the business service.

Business service operations and technical service operations allow for detailed information about how the business service will be provided by an application or component in order to realize a business function. The operation describes the concrete availability of the associated business services provided by applications or components. For example, to realize the business function Invoicing, the operations Invoicing through B2B Marketplace, Invoicing with Credit Check, Invoicing without Credit Check, and Invoicing through EDIFACT are required. Additionally, parameters and return values can be defined for the operation in order to describe the interface of a service call as well as provide useful information to drive development activity. The operation is defined in the *Business Function Operations Page View* for the specific business function that the operation helps to realize via the business service. Orchestrations that specify the process whereby operations call one another in order to carry out the business service can be defined in the *Orchestration Page View*.

In some enterprises, business functions may be bundled in one or more functional modules in order to capture specific business requirements that are or can be fulfilled by an application. The functional module is owned by a domain.

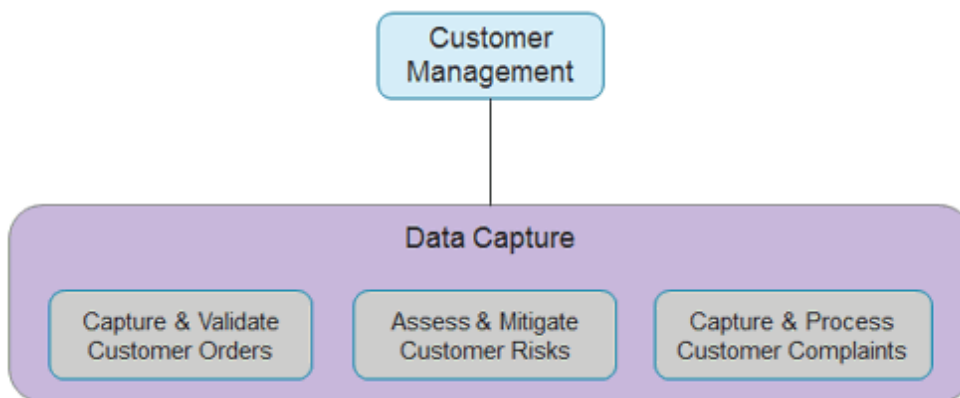


FIGURE: Business functions assigned to a functional module

In the example, the functional module Data Capture bundles the business functions Capture and Validate Customer Orders, Assess and Mitigate Customer Risks, and Capture and Process Customer Complaints. In this sense, a functional module can be considered an application prototype that captures functional aspects necessary to the enterprise architecture. A functional module can be created for a domain in the *Functional Modules Page View* and the business functions can be assigned to a functional module in the *Business Functions Page View*.



Please note that for enterprises that structure their business architecture by means of capabilities, business functions represent a concrete business activity and must be located at the leaf-level of the domain/business capability hierarchy. This is described in more detail in the section [Documenting the Enterprise's Business Capabilities](#).

Defining the Operational Business Support for a Domain

An operational business support is the active or planned support that is currently provided as a result of ongoing development or roll-out activity.

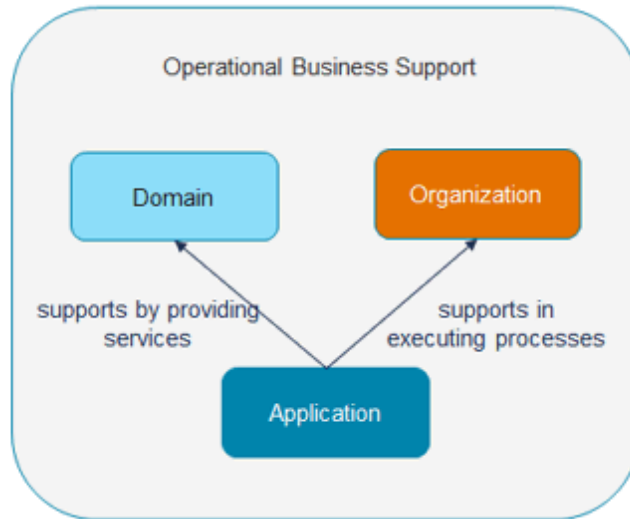


FIGURE: Operational business support of a domain

The operational business support consists of three dimensions:

- The provider of the operational business support. In the example above, an application provides the business support
- The domain that the application supports by providing business services that the domain requires to fulfill a business function.
- The organization that is supported by the application. The application typically supports an organization in its activities to realize business functions associated with a domain.



Please note that the business support concept in your enterprise may differ. Business supports are configured by your solution designer. Depending on the configuration, the business support provider may be either an application or organization. The business support provider may provide support to either a domain or business process for either an organization or market product.

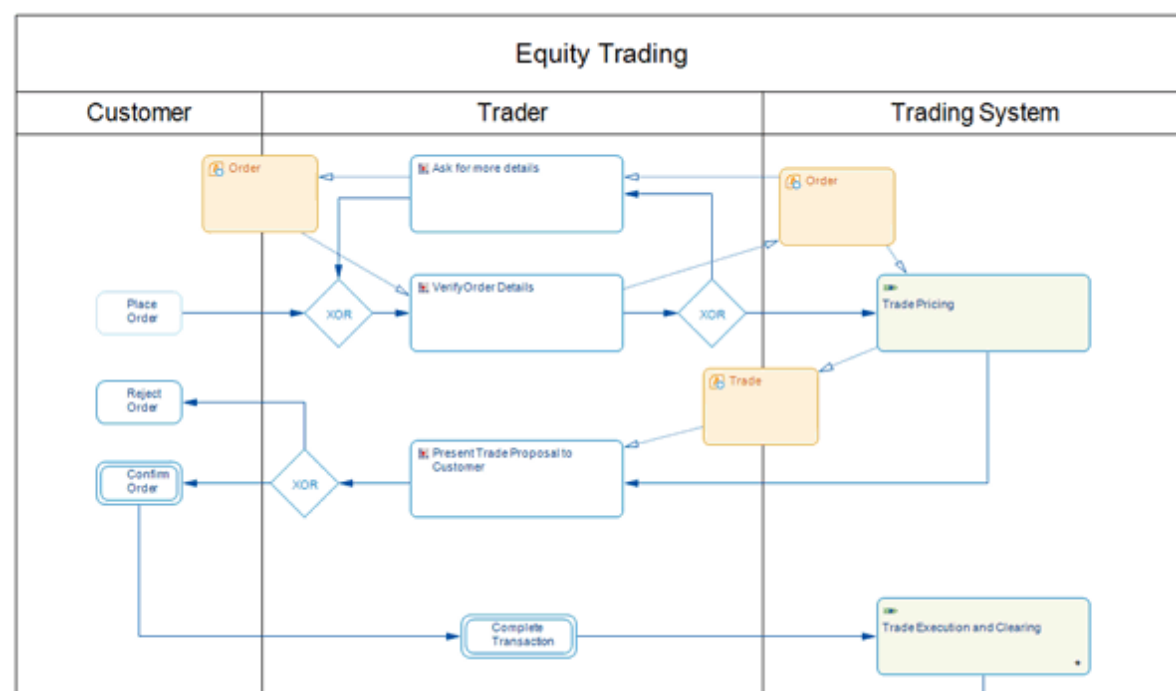
The operational business support is defined in the *Business Support Page View* for the domain. Once operational business supports have been defined, you can review the support provided by the application. The *Business Support Map Report Page View* provides an overview of the domains and organizations supported by the application Trade*Net (orange cell). The reports displays all operational business support provided by all applications for the set of domains and organizations that the application Trade*Net supports.



Once you have captured the business support that supports your domains, you can analyze the current operational support and plan your enterprise's medium-term and long-term IT. In the context of an IT strategy, you can roadmap your company's long-term strategic business support by means of the Business IT Synchronization capability. In the context of a master plan, you can roadmap the medium-term tactical business supports by means of the Target Architecture Definition capability. Both capabilities are part of the IT Planning Basic sales package.

Documenting the Service Architecture for a Business Process

Software AG provides a means to design diagrams that allow you to model the service architecture for a business process or an application. You can design the diagram to visualize all relevant dependencies between the subordinate business processes as well as the business services requested that are by a business process or provided by an application.



The design of service diagrams in Alfabet follows BPMN conventions. A service diagram may contain events or activities, simple connectors such as sequence flows, message flows, or associations, complex connectors such as gateways as well as boundary elements such as pools or swim lanes. In addition, business objects, business services, and business functions can also be visualized in a service diagram. Additionally, links to other service diagrams can be defined for the activities, business services and business functions displayed in the service diagram.

Service diagrams are designed in the *Service Diagrams Page View* available in the object profile of a business process or application. To design a diagram, you must have access to the tool Alfabet Diagram Designer. For more information about designing diagrams in Alfabet, see the reference manual *Designing IT Landscape Diagrams in Alfabet*.



Please note that business process modeling is also available via ARIS – Alfabet Interoperability Interface. To access this capability, your enterprise requires the necessary licenses as well as installation of ARIS – Alfabet Interoperability Interface. For more information, see the reference manual *ARIS – Alfabet Interoperability*.

Analyzing the Domain Model

A variety of reports are available to help you understand the business activities supported by each domain in the domain model. In addition, your enterprise may configure reports that are specific to the analysis needs of your company. The following provides a few of the standard reports that are available:

- *Standard Domain Application Diagram*: This view allows you to analyze a view a diagram defined by your enterprise displaying domains and business functions. Indicators can be superimposed on the domain landscape to provide qualitative information about a set of domains. For example, you could assess and compare the cost efficiency or operational excellence for a defined set of domains and their associated business functions.
- *Domain Interrelationships Page View*: The concept of domains aims to achieve a partitioning of the business architecture into disjoint domains, whereby each domain owns a set of relevant business functions, business objects, business processes, applications, ICT objects, components, standard platforms, or vendor products). The report helps you to identify and reduce interdependencies between domains by showing all objects assigned to a selected domain that have a different primary domain or other associated domains. Based on the results of this report, you can determine which domain is the correct domain for each object and, if necessary, what action to take to reduce ambiguous object assignment across domains.
- *Service Provider Lifecycle Report Page View*: This Gantt chart displays the scheduling of the business services defined for a domain's business functions and provides you with information about which IT services are adequately supported over time to realize the enterprise's business functions. You can view the business services that are actually being provided or may potentially be provided by applications by means of operational business supports.
- *Application Evaluation Report Page View*: This report displays aggregated indicators for all applications defined as primary objects for a selected domain. Various prioritization schemes bundling indicators can be select to show qualitative information such as the criticality, reliability, or alignment of the applications in the domain. The content of this report will depend on the evaluation types and prioritization schemes that have been configured by your enterprise.
- *Functional Modules Page View*: This report helps you to understand the business functions that are bundled in the functional modules assigned to the selected domain. You can assess which functional aspects may be missing or unnecessary for the functional modules and thus perform the restructuring needed to adequately fulfill the specific business requirements of your enterprise.

Documenting the Enterprise's Business Capabilities

Some enterprises define their business by means of business capabilities. This approach helps the enterprise to understand which business capabilities the enterprise needs to operate and which business capabilities are most important for business success. An analysis of the enterprise's business capabilities allows the synergies, redundancies, and pain points in the enterprise to be identified so that later projects can be formulated that will ultimately align the business capabilities with the business strategy. Business capability management helps to bridge the gap between the business and IT and establish a framework for investment decisions.

A business capability is an abstract description of what is done in an enterprise to meet its business objectives. It describes what a business does independent of the organization's structure, business processes, people or domains. In contrast to the business process view, which asks how the business is run, a business capability describes the main functions and competencies that describe what is necessary to run the business.

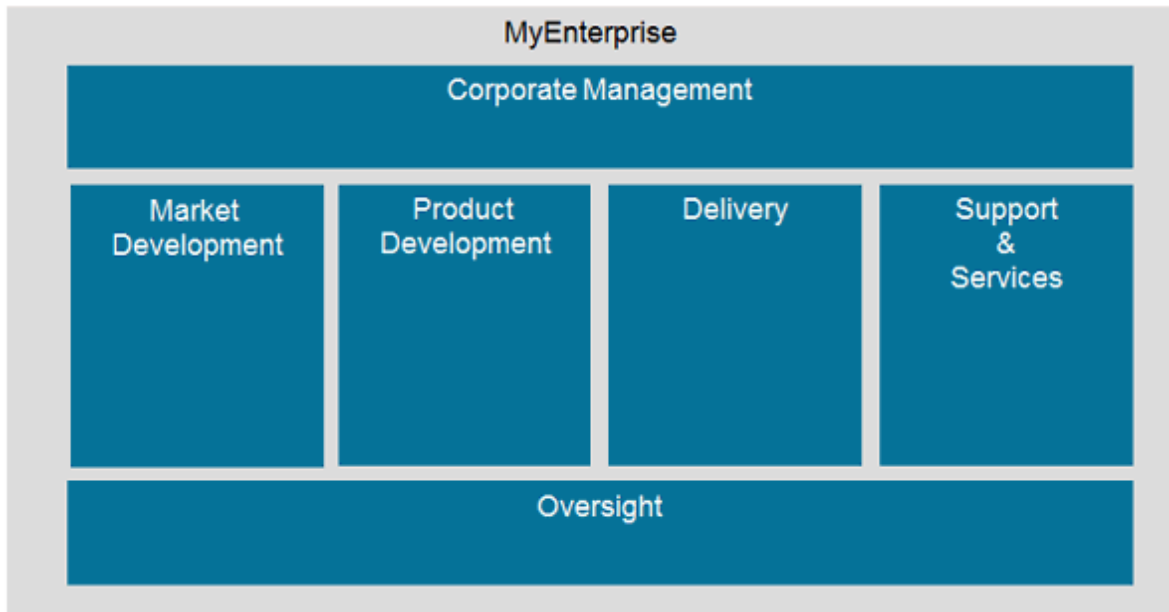


FIGURE: Six major business capabilities in an enterprise

The enterprise's main business capabilities should be defined as the highest level business capabilities. It is recommended that this is limited to 5-10 generic business capabilities. Each business capability may have multiple subordinate business capabilities, whereby it is recommended that the hierarchy not have more than 3 or 4 levels to provide manageability. A business function should be defined at the leaf-level of each business capability in order to describe the business activities relevant to the business capability.

There is no explicit class for business capabilities in Alfabet. Business capabilities are initially captured as domains in Alfabet.

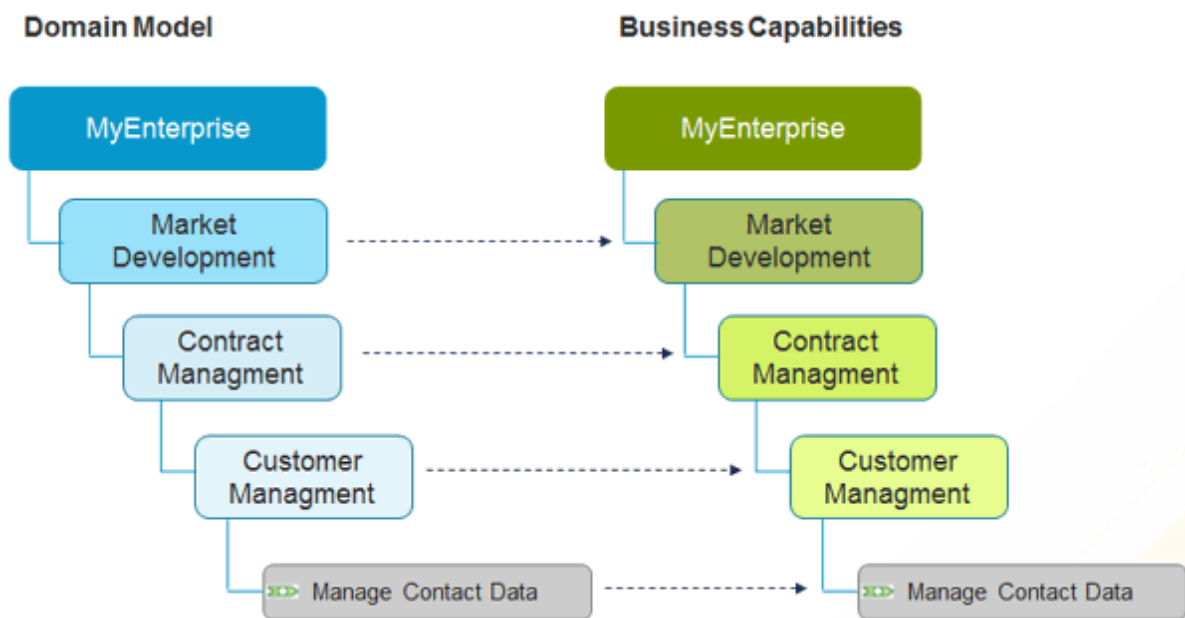


FIGURE: Each domain in the domain model is a business capability in the business capability map

Each domain in the domain model represents a business capability in the context of the **Business Capability Management** functionality. The hierarchy of domains and sub-domains is thus represented as a

hierarchy of business capabilities and subordinate business capabilities with business functions at the leaf-level of the hierarchy.

All business capabilities are targets for evaluation. Guidelines to evaluate an enterprise's business capabilities are defined in a business capability map whereby, the performance of the business capabilities and their business functions can be assessed according to various evaluation types and capability aspects, which allow different perspectives to be captured. One or more users can be specified to assess the business capabilities. Data is gathered for the business capabilities and business functions via evaluation types and an evaluation summary is derived. The assessment allows the organization to determine current performance vs. desired performance and gain a clear understanding of which business capabilities are core to the business and warrant further IT support.

Thus business capabilities can be consolidated based on a common understanding of which areas of activity are associated with one another as well as where the gaps and deficiencies in performance lie. The business capability map provides a reliable platform for discussing business needs and for mapping IT support in order to support an understanding between the business and the IT, help to prioritize demands and projects, and expose how costs can potentially be optimized.

The following information is available:

- [Prerequisite: Configuring the Business Capability Management Capability](#)
 - [Configuration of Domain Stereotypes](#)
 - [Configuration of Evaluation Data](#)
- [Defining Business Capabilities and Creating a Business Capability Map](#)
- [Evaluating the Business Capabilities](#)
- [Analyzing the Enterprise's Business Capabilities](#)



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

Prerequisite: Configuring the Business Capability Management Capability

In order to work with the business capability management capability, the following configuration is required:

- [Configuration of Domain Stereotypes](#)
- [Configuration of Evaluation Data](#)

Configuration of Domain Stereotypes

The business capability management capability requires that a domain model is configured as described in the section [Prerequisite: Configuring the Domain Management Capability](#). When you conceptualize the domain stereotypes to implement in the business management capability, you should consider that the higher-level business capabilities describes business activities broadly, with each subordinate level in the hierarchy becoming more granularity in detail.

Business functions must be available for the domain stereotype at the leaf level in the domain hierarchy in order to allow concrete business activity to be defined and evaluated. Please note that this requires that business functions are configured as permissible for the lowest-level domain stereotype. Your solution designer will configure the domain management capability via the configuration tool Alfabet Expand. For more information, see the section *Configuring Domain Models and Domain Planning* in the reference manual *Configuring Alfabet with Alfabet Expand*.

Configuration of Evaluation Data

In addition to configuring domain stereotypes in the configuration tool Alfabet Expand, the following must be configured in order to evaluate business capabilities:

- In order to evaluate business capabilities, evaluation types (and their indicators) must be configured in the **Configuration** module. A user in your enterprise with access permissions to the **Configuration** module will typically configure evaluation and reference data that will allow the user community to evaluate and analyze the business capabilities. Your enterprise can determine the KPIs relevant to assess the business capabilities. The indicators that are to be evaluated will depend therefore on the goals of the evaluation of the business capabilities. For example, you may want to evaluate the market differentiation of the business capabilities or their revenue contribution or criticality to the business. In the case of an evaluation type Market Excellence, for example, the indicators customer Excellence, People Excellence, Process Excellence, and Product Leadership could be assessed.

Please note that the evaluation types that will be used in the context of business capability management must be assigned to the object class Domain in the **Configuration** module. For more information, see about the configuration of evaluation types, see the chapter *Configuring Evaluations, Prioritization Schemes, and Portfolios* in the reference manual *Configuring Evaluation and Reference Data in Alfabet*.

- A summary evaluation type must be assigned to the business capability map. The summary evaluation type is necessary to compute a final value for each business capability as well as a delta value for the current and planned state. The summary evaluation type is typically defined in a computation rule.

Defining Business Capabilities and Creating a Business Capability Map

Before you begin the actual process of capturing business capabilities and preparing the evaluation, you should have a general idea what the business capabilities are in your enterprise, what questions you want to answer in the evaluation, and which level in the business capability hierarchy will be assessed. For a company-wide assessment, for example, you might want to assess the second level in the business capability hierarchy and examine only lower levels in detail if they require further attention. It is also possible to assess only specific business capabilities, although these should be evaluated at the same level in the business capability hierarchy in order to execute comparative analyses later.

Once the business capability map is created, you must you define the "rules" for the evaluation -- who may be an evaluator and what issues will be assessed. The level in the business capability hierarchy that will be assessed usually informs who the evaluators will be as well as the indicators to be evaluated. Typically, multiple evaluation groups will evaluate each capability but the evaluation owner is ultimately responsible for assessing their input and assigning a final value to a business capability.

The following steps are recommended to document and analyze the business capabilities in your enterprise:

- 1) **Conceptualize business capabilities:** Before you can begin to work with the *Business Capability Maps Explorer*, you should first conceptualize the structure of the functional architecture in your enterprise. The business capabilities should answer the question: What is necessary to run the business? They should be defined in a way that indicates their business purpose and they should be defined independently of an organizational structure, business process model, or IT-specific functionality. It is recommended that you conceptualize 5-10 generic business capabilities at the top of the business capability hierarchy. Each business capability may have multiple subordinate business capabilities, whereby it is recommended that the hierarchy not have more than 3 or 4 levels to provide manageability. A business function should be defined at the leaf-level of each business capability in order to describe the business activities relevant to the business capability.
- 2) **Capture business capabilities as domains:** Once you have conceptualized the business capabilities and business functions in your enterprise, you must capture each business capability as a domain in the *Domains Explorer*. All domains are potential targets of the evaluation of business capabilities.
- 3) **Enumerate business capabilities:** Once all business capabilities are captured as domains, you should enumerate the business capabilities (domains) in order to provide consistent numbering of the hierarchy when working with a business capability map. To do so, click the root node of the **Domain** explorer and click **New > Enumerate Domains**. Each domain (business capability) at the lowest level of the hierarchy should have one or more business functions defined that describe the business activity of that business capability.
- 4) **Create the business capability map:** Once the hierarchy of business capabilities is captured in the **Domain** explorer, you can create a business capability map in the *Business Capability Maps Explorer*. A business capability map allows you to define the guidelines to evaluate the business capabilities defined for the enterprise. The performance of business capabilities and business functions can be evaluated according to specific business capability aspects and evaluation types by a defined group of evaluators. Only one business capability map may be created. When you create the business capability map in the **Business Capability Map** explorer, the **Business Capability Map** editor will open. Please note the following:
 - The business capability map requires a name.
 - You should provide a description of the business capability map so that other users understand its purpose.
 - As the creator of the business capability map, you are automatically defined as the authorized user per default. The authorized user of the business capability map can be changed in the **Authorized Access** tab. You can also define any user groups that should have Read/Write access permissions to the business capability map in the **Authorized Access** tab. Please note that mandates are not supported in the context of the business capability map.
 - In the **Evaluation** tab, define the attributes for the business capability map including the summary evaluation type (a computation rule) and an icon group to visualize the computed delta indicators (optional).

Business Capability Map

Basic Data
Evaluation
Authorized Access

Summary Evaluation Type

Visualization of Delta Indicator

	Icon	Min. Value	Max. Value	Caption
1		<input type="text"/>	<input type="text" value="-2.5"/>	<input type="text"/>
2		<input type="text" value="-2.5"/>	<input type="text" value="-1.5"/>	<input type="text"/>
3		<input type="text" value="-1.5"/>	<input type="text" value="-0.5"/>	<input type="text"/>
4		<input type="text" value="-0.5"/>	<input type="text" value="0.5"/>	<input type="text"/>
5		<input type="text" value="0.5"/>	<input type="text" value="1.5"/>	<input type="text"/>
6		<input type="text" value="1.5"/>	<input type="text" value="2.5"/>	<input type="text"/>
7		<input type="text" value="2.5"/>	<input type="text"/>	<input type="text"/>
8		<input type="text" value="9999"/>	<input type="text" value="9999"/>	<input type="text"/>

- 5) **Specify evaluation groups:** Evaluation groups are the user groups in the enterprise that may be required to participate in the evaluation. In some cases, this might be the owners of the business capability, but other perspectives such as IT management, business analysts, or the enterprise architecture team may also be beneficial. For example, you may want to have domain experts, strategic planners and IT architects evaluate the business capabilities in order to examine them from various perspectives and stakeholder groups. Evaluation groups should be added that include any user that will be required to evaluate a business capability or who may be considered the evaluation owner. You can define one or more evaluation groups that are responsible for evaluating the business capabilities in the *Business Capability Map Evaluation Groups Page View*. All evaluation groups that you define will be displayed below the business capability map node in the **Business Capability Map** explorer.

- 6) **Specify an evaluation owner for each business capability:** The evaluation owner is the evaluation group that is primarily responsible for the evaluation of the business capability. If there are significant differences in the values assigned by the various evaluation groups, the evaluation owner can initiate a discussion with the other evaluation groups about the values assigned. The evaluation owner is responsible for determining the final value that will be used in the various reports available for the business capabilities. You can assign an evaluation owner to each business capability assessed in the evaluation in the *Business Capability Map Evaluation Owner Page View*. Please note that all potential evaluation owners must first be assigned to the *Business Capability Map Evaluation Groups Page View*.

- 7) **Specify the indicators that will be assessed:** Typically, evaluation types will be configured by your enterprise and should address the dimensions that are relevant to your business. The evaluation types should address questions that are generic across all business capabilities so that the business capabilities can be compared at the analysis stage. You can define the evaluation types that will be used in the evaluation of the business capabilities in the *Business Capability Map Evaluation Types Page View*.

- 8) **Specify the perspectives that will be evaluated:** Each business capability will be evaluated in the context of an evaluation type and from the perspective of a business capability aspect. A business capability aspect represents a specific perspective from which the performance of the business capability is evaluated. You can specify capability aspects based on predefined aspect types such as the Organization, Product, Custom, Business Segment, and Business Line. You can assign the business capability aspects to be applied to the evaluation of business capabilities in the *Business Capability Map Aspects Page View*.

Evaluating the Business Capabilities

The business capability evaluation is the process of getting structured top-down business input about the satisfaction and future expectations of the current IT support. Once the business capability map has been specified, each relevant business capability must be evaluated by the relevant evaluation groups. Typically, the evaluation will be down by the evaluation owner, who is responsible for the business capability, as well as all other evaluation groups, who will contribute their input based on their perspective as an evaluation group. Once all evaluation groups have provided their input, the values defined for each business capability must then be consolidated to one final summary value.

The following steps are recommended to evaluate the business capabilities:

- 1) **Evaluate business capabilities:** The values for the evaluation are collected from each evaluation group in the *Business Capability Map Evaluation Page View*. You should determine the level of the business capability hierarchy that should be assessed company wide. For example, you could require all evaluators to assess the second level and then drill-down to lower levels in the hierarchy if a business capability requires more detailed attention. Not all business capabilities must be assessed; it is possible to assess only those that currently require attention. The evaluation groups should evaluate the business capabilities for each evaluation type according to at least one business capability aspect. Please note the following:
 - Each evaluation group must evaluate its business capabilities.
 - Each evaluation group will typically be required to evaluate all business capabilities of all other evaluation groups defined in the business capability map.
- 2) **Compute and consolidate indicators:** The multiple values defined for a business capability are then computed, reviewed, and consolidated by the business capability's evaluation owner in the *Business Capability Map Consolidation Page View*. The evaluation owner of the business capability must determine the final value that will be used in the computation of values displayed. The following procedure is recommended when working with the *Business Capability Map Consolidation Page View*.
 - The coordinator of the business capability management process should compute or update the indicators for all or some of the business capabilities displayed in the table. The computation is based on the summary evaluation type defined for the business capability map. The computed values are for the business capability aspect for which the values were defined.
 - The evaluation owner can review the values defined for all evaluation types that have been made by other evaluation groups. Typically, the evaluation owner would initiate a discussion about conflicting values with the other evaluation groups.
 - If necessary, an evaluation owner may edit and adjust the values that he/she has assigned to an evaluation type for the business capabilities that he/she is the owner of. Please note that a member of an evaluation group that is not the owner of the business capability can only view

the indicators computed for the business capabilities in the business capability map. To edit any values, he/she must return to the *Business Capability Map Evaluation Page View*.

- An evaluation owner can review and consolidate the values that have been defined by all other evaluation groups for a business capability that he/she is the owner of. Once any outstanding conflicts in the evaluation of the business capabilities have been discussed (and, if necessary, resolved), the evaluation owner can define a single consolidated value for each evaluation type of the business capability. The values can be consolidated across evaluation groups and/or business capability aspects. This must be done for the current state and the planned state
- The coordinator of the business capability management process should generate the delta values that represent the difference between the current value and planned value. The delta values indicate the areas in the business that require attention.

Analyzing the Enterprise's Business Capabilities

Once the business capabilities have been evaluated and the evaluation consolidated, you can execute various analyses in order to understand the business capabilities that require attention. The *Business Capability Map Summary Page View* allows you to view a summary of the consolidated values of the business capabilities. You can determine the relative importance of each evaluation type which is applied to the relative ranking of the business capabilities. You can also view values for the current state or planned state, or view the delta value representing the difference between the current and planned state. A business capability with an especially high delete is typically a business capability that requires attention. The following provides a few of the standard reports that are available for a selected business capability displayed in the *Business Capability Map Summary Page View*.

- *ICT Support for Capabilities Page View*: This report displays the organizations and ICT objects providing the business supports that are associated with the selected business capability's leaf-level business functions. The number of business services provided by the ICT object and the number of business supports that use the business service are shown. Business services that are not used by a business support may be potential candidates for discontinuation.
- *ICT Object Services Usage Page View*: This report displays the applications providing the business service, the business functions realized by the business service, and any other applications providing business services that also realize that business function. In this way, you can determine which business services may be redundant and therefore potential candidates for discontinuation.
- *Capability Architecture Diagram Page View*: This report displays the business processes and ICT objects associated with one or more selected capabilities in order to see whether changes are required on the process-side or the IT-side of the architecture.
- *Capability Function Usage Report Page View*: This report displays the use of the ICT objects in providing business services for the associated business functions (leaf-level capabilities). The bar chart indicates how many business services are provided by the ICT objects and how many of those business services are actually used by business supports. A difference in the number of business services offered and the number of business services used indicates that action should be taken in regard to the business capability.
- *Capability ICT Redundancy Report Page View*: This report displays the number of ICT objects providing business services for the business functions associated with a selected capability over a two-year period.

- *Business Support Map Report Page View:* This report allows you to understand which business services are unused or redundant in the context of a business support matrix.
- *Capability ICT Cost Report Page View:* This report allows you to understand the distribution of business service costs for the applications providing the business services associated with a selected business capability.
- *Project Report Page View:* This report allows you to understand the running projects that are associated with a selected business capability. Please note that to capture projects in Alfabet, you must have access to the Project Portfolio Governance capability which is part of the IT Planning Advanced sales package.
- Service products can be evaluated in the enterprise in order to understand specific dimensions of a service product's performance. The evaluation types that your enterprise configures for the class Service Product can be evaluated in the Evaluation page view available for the service product. Portfolio analyses of the evaluated service products can be viewed based on configured portfolios in the Service Product Portfolio page view for a service product group and in the Service Product Groups explorer or a service provider in the Service Product Catalog explorer.

Capturing Business Appraisals in the Enterprise

A business appraisal is a means for an enterprise to gain an initial understanding of the alignment between the business and the supporting IT. By means of a business appraisal, the support available to as-is or to-be architecture can be assessed in terms of the support provided for a set of business processes and organizations. The advantage of capturing this information via business appraisals is that the definition of business supports is not required to conduct the business appraisal. Business appraisals can be implemented to understand the current level of support or as the a basis for taking action and planning medium-term tactical business support or long-term strategic business support for the enterprise.

The business appraisal is an evaluation of the as-is and to-be support for a business process for a selected organization in regard to a specific operational aspect. Each business appraisal should be evaluated based on indicator types configured for your enterprise. For example, indicator types providing evaluations of the as-is IT support, the to-be IT support, the need for action, and the importance of the relevant business process for the enterprise could be configured for the evaluation.

Business appraisals are defined in the context of a business support map defined for an IT strategy or master plan. The business appraisals are displayed as a matrix object labelled **Business Appraisal** with an icon representing the indicator. For example, colored bubbles for the indicator type Need for Action might be displayed in order to understand the general level of concern regarding the support to the as-is or to-be architecture. Once the business appraisals have been evaluated, the gaps between the to-be and as-is support can be assessed allowing the enterprise to determine priorities for an IT strategy in order to align the business and IT as well as future projects to realize the IT strategy.




A business support map is the graphic visualization of the business support provided by specified objects in order to support the planning of the short-term, medium term, and long-term to-be architecture. The map is visualized as a matrix with an X-axis and a Y-axis. The X-axis of a standard business support map will display business processes and the Y-axis will display organizations that are supported. In some industry segments, it is more relevant to analyze the business support for market products than for organizations. If this is the case, market products may be configured for the Y-dimension of the business support. Furthermore, some enterprises may describe business supports to provide support to domains or business capabilities of the business rather than business processes. If this is the case, domains may be configured for the X-dimension of the business support. For more information about working with business support

maps, see the section *Appendix: Working with Business Support Maps* in the reference manual *IT Planning Basic*.

The following configuration is required to work with business appraisals:

- Operational aspects must be defined for the business processes displayed on the X- axis and the organizations displayed on the Y-axis of the matrix. This is carried out in the *Operational Aspects Page View* for the respective business process and organization.
- Evaluations types and their indicator types must be created in the **Evaluations and Portfolios** functionality and assigned to the object class **Business Appraisal** in the **Class Configuration** functionality. For more information about configuring evaluation types and indicator types, see the chapter *Configuring Evaluations, Prioritization Schemes, and Portfolios* in the reference manual *Configuring Evaluation and Reference Data in Alfabet*.
- The indicator type to display on the business appraisals in the business support matrix must be specified in the XML object **ITMapDef** in the configuration tool Alfabet Expand. For more information, see the section *Configuring Standard Business Support Matrices* in the reference manual *Configuring Alfabet with Alfabet Expand*.

The following is required to create and evaluate business appraisals

- Create a master plan and master plan map as described in the chapter *Target Architecture Design* in the reference manual *IT Planning Basic*.
- Specify the business processes and organizations that you want to assess. In the *Business Support Map Page View*, add the relevant business processes to the X-axis and the organizations the Y-axis. For more information about defining the axes of the business support map, see the section *Appendix: Working with Business Support Maps* in the reference manual *IT Planning Basic*.
- Set-up the business support map to view relevant data. In the *Business Support Map Page View*, click **View Options** in the toolbar and select **Show Business Appraisals**. It is also useful to visualize the operational aspects relevant to the business appraisal. To do so, select **Show Attributes** in the **Details** tab.
- Create business appraisals for the business processes and organizations. To do so, click the **Business Appraisal**  button in the toolbar and click in the matrix cell corresponding to the business process and organization that you want to evaluate.
- Evaluate each business appraisal. To do so, select a business appraisal in the business support map and click **Details > Navigate to Inventory Object**. In the object profile of the business appraisal, open the *Evaluation Page View* and evaluate the indicator types.
- Assess where gaps exist in the as-is or to-be support. To do so, return to the *Business Support Map Page View*. Icons configured for the indicator types will be displayed on the business appraisal.