Enterprise Architecture Management Tool Survey 2008

Prof. Dr. Florian Matthes, Sabine Buckl, Jana Leitel, Christian M. Schweda

Software Engineering for Business Information Systems (sebis)
Ernst Denert-Stiftungslehrstuhl
Chair for Informatics 19
Technische Universität München

Boltzmannstraße 3, 85748 Garching b. München, Germany

eamts@systemcartography.info

Main Sponsors

Co-Sponsors

About sebis

sebis is the chair for *Software Engineering for Business Information Systems* at the Institute for Informatics of the Technische Universität München. *sebis* has been established in 2002 with funding of the Ernst Denert-Stiftung and is headed by Professor Dr. Florian Matthes. The main research areas of *sebis* are:

- Software Cartography: Development of multi-faceted and formal models that help to manage (plan, build, operate, optimize) complex software application landscapes consisting of hundreds or thousands of information systems.
- Innovative technologies and software architectures for enterprise information and knowledge management (enterprise solutions, groupware and social software).
- Domain-specific and reflective languages and models for families of business applications.

sebis is using software engineering methods (model construction & abstraction, analysis & design, construction & evaluation) and is working in close relationship with industrial partners and with organizations from the public sector.

Professor Matthes puts particular emphasis on the knowledge transfer from academia to industry. For example, he is co-founder of CoreMedia AG, of infoAsset AG, and of 20six Web log services AG, which at present employ a total of approx. 150 employees.

ISBN-13: 978-3-00-024520-6

Abstract

The number of application systems supporting various business processes, and the geographical and organizational distribution of these systems together with their interdependencies have lead to complex structures in today's companies. Enterprise architecture (EA) management targets these complex structures and is concerned with the *alignment of business and IT*. EA management therefore not only deals with the information technology but also with business processes, organizational structures, strategies, business capabilities, projects, etc.

To manage the evolution of the EA, companies started to document the artifacts mentioned above, especially their interaction and dependencies. The high amount of information to be gathered for managing the EA increases the need for tool support, enabling a collaborative management process involving users with different roles. An appropriate tool must not only capture relevant information, it must also process this information, using e.g. reports or visualizations supporting the users when performing different EA management tasks.

This survey analyzes major players in the emerging market of EA management tools and gives a brief introduction to typical tasks in EA management using a scenario-based approach. The scenarios and the list of criteria were developed in cooperation with sponsors and partners ranging from medium-sized to large companies acting in different markets.

Each tool is analyzed by simulating different scenarios and analyzing specific functionalities necessary for EA management. The evaluation of each tool is categorized into two parts: One part deals with specific functionality, e.g. *adapting the information model, supporting multiple users and collaborative work, creating visualizations of the application landscape, or usability.* The second part analyzes the tools' enterprise architecture management support, e.g. *landscape management, project portfolio management, application architecture management, SOA transformation.* Using this evaluation structure, readers can assess how well a particular tool fulfills their individual requirements.

Acknowledgments

We thank the sponsors and partners of the *Enterprise Architecture Management Tool Survey 2008* for their financial support and for sharing their insights into EA management with us, greatly influencing the list of criteria and the scenarios this survey report is based on.

We also thank all participating tool vendors for their support and open information policy. We would like to express our gratitude for their patience and their readiness to provide training and support on short notice.

Last but not least we want to thank Thomas Dierl and René Ramacher for their careful and detailed support simulating and evaluating the scenarios with the tools provided. We would also like to thank the other *software cartography* team members at sebis, Alexander Ernst and Josef Lankes, for their reviews and comments on the text.

Garching b. München, April 2008

Prof. Dr. Florian Matthes
Sabine Buckl
Jana Leitel
Christian M. Schweda

Contents

CHAPTER 1

Introduction

Contents

The *Enterprise Architecture Management Tool Survey 2008* (EAMTS2008) evaluates the solutions of major players in the market of enterprise architecture (EA) management tools. The survey is the successor of the *Enterprise Architecture Management Tool Survey 2005* [se05] performed by sebis. EAMTS2008 comes up with enhanced and new scenarios as well as a revised list of criteria – based on our experience and the input of the sponsors and partners of the new survey. Nine tools of major players in the EA management tool market are evaluated in detail, while four additional ones are shortly sketched.

1.1. Motivation & Participating Sponsors and Partners

The complexity of the information technology (IT) used in enterprises has grown during the last decades. IT started as a support function becoming more and more relevant for business by increasing efficiency. Today, IT also plays the role of an enabler making it possible to put new strategies and processes into practice, but it may also represent a barrier in the realization of opportunities. This growing role of IT has led to the awareness that business and IT have to be managed as a whole.

The high number of application systems in the application landscape of an enterprise and the multitude of interdependencies between IT and business makes the EA management (EAM) process increasingly difficult to execute without tool support. Many vendors offer EA management tools and the variety of products is remarkable. One reason for the diversity is certainly the origin of the vendors from different areas (companies come e. g. from testing, process modeling and metamodeling), another reason might be given by different opinions what EA management is and what it includes. Thus, a plethora of different approaches to EA management exists, each approach with its individual strengths and weaknesses.

This tool survey is intended for companies introducing an EA management tool as well as companies that are currently engaged in EA management endeavors. On the one hand, the tool survey provides a profound introduction to EA management and lists best practice scenarios that aggregate the experience of our sponsors and partners. These best practice scenarios were built based on the practical examples for tasks provided by the participating companies. Based on this input and on our research, the survey can be seen as a guideline to the various parts of an EA management. On the other hand, it offers a structured decision support for companies searching an adequate EA management tool by detailing how a particular tool manages to address different areas and different processes related to EA management. Furthermore, because the list of criteria and all scenarios used for evaluation have been developed in cooperation with our sponsors and partners, companies that are interested in the field can use this survey as a guideline how to perform EA management. The study also represents the current requirements demanded by users. Therefore, this tool survey can be seen as a link between tool users and tool vendors.

The evaluation does *not* lead to a simple ranking of the tools but to a *scorecard* based on the criteria and the scenarios. Companies can match their individual requirements with the scorecard, in order to find out which tool best satisfies their specific needs. Tool vendors will benefit from a comprehensive list of requirements, formulated in cooperation with industrial users, as well as from the evaluation that might help refining the further development of the tools.

The industry partners participating in this survey are divided in three different groups, *main sponsors*, *co-sponsors*, and *partners* as shown in Figure 1.1. We like to express our gratitude to all of our industry partners for supplying us with their insights from practice, which makes this survey a comprehensive overview on the state of the art in executed EA management. Our special thanks goes to our sponsors, which made the survey possible by their financial contributions.

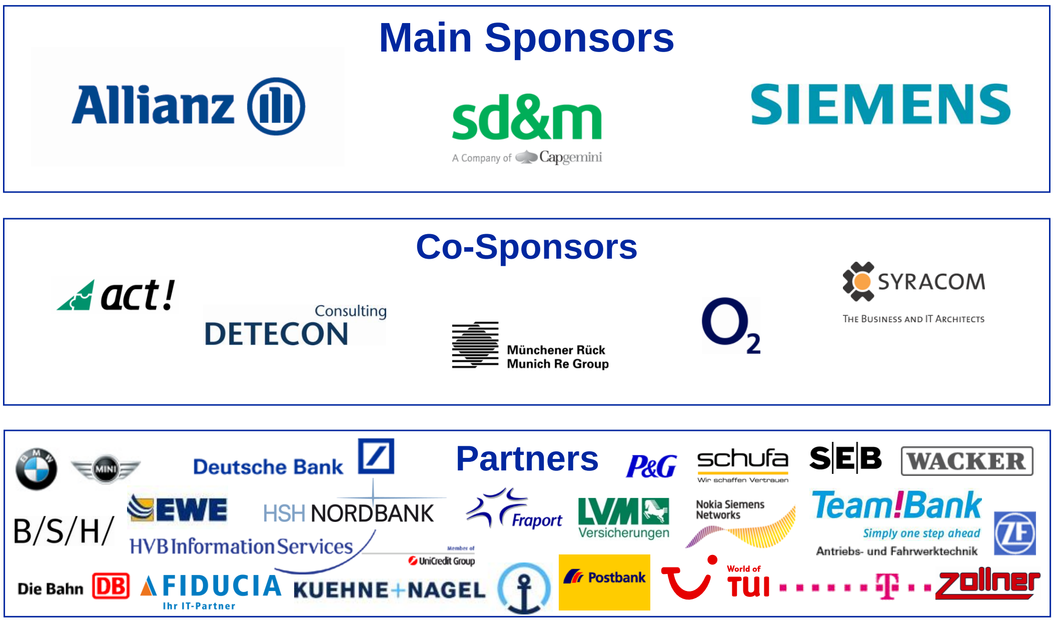


Figure 1.1.: Participating sponsors and partners of the EAMTS2008

1.2. Who we are

sebis is the chair for *Software Engineering for Business Information Systems* at the Institute for Informatics of the Technische Universität München. *sebis* is partially funded by the Ernst Denert-Stiftung and headed by Professor Dr. Florian Matthes.

In our research project *Software Cartography*, which started in February 2003, we develop methods and modeling techniques for documenting, evaluating, and planning application landscapes. Thereby, we employ notations and visualization techniques utilizing concepts from the field of conventional cartography, e. g. we have introduced the concept of *Software Maps* visualizing the application landscape and thereby showing parts of the EA.

In our current research activities, which originate from the software cartography project, we cooperate with several industry partners to work on best practice management methodologies and intuitive visualizations for application landscapes. These application landscapes, which are complex systems consisting of hundreds or even thousands of application systems, are a major investment of modern enterprises and are continuously changed to support new business goals and business processes. The documentation, evaluation, and planning of these application landscapes are major challenges of enterprises, especially when seen in the context of alignment of business and IT.

In order to provide methodologies and visualizations valuable in this context, we try to address management concerns ranging from the business architecture evolution via the application architecture to the technical infrastructure, on which the applications are planned, built, and operated. These dynamic aspects covering the evolution of the application landscape are especially taken into consideration, e. g. via projects affecting entities of the application landscape, linked to strategies and goals to which these elements should be aligned.

The following list provides an overview about ongoing research projects at *sebis*:

- Software maps are used in several enterprises as a means for managing the application landscape. Nevertheless, these visualizations are often created manually in a labour expensive and error prone process. An approach for automatically generating software maps from information about the application landscape can therefore be seen as a valuable contribution to managing the application landscape. In our research project *Software Cartography Tool (SoCaTool)*, we have developed such an approach and show its applicability by implementing a prototypic tool.
- The *Enterprise Architecture Management Pattern Catalog (EAMPC)* builds on our research about EA management viewpoints and information models in order to provide best practice building blocks that aid in the construction of an approach to EA Management. Therefore, the EAM pattern catalog provides concern-specific methodologies (M-Pattern), viewpoints (V-Pattern) and information model patterns (I-Pattern), which can easily be reused in the creation of an organization-specific approach to EA management.
- Considering the complexity and importance of the application landscape as a critical asset in modern organizations, its management should be facilitated by decision support techniques. These techniques allow prediction and evaluation of effects caused by the evolution of an application landscape in a certain direction. In the research project *Metrics for Application Landscapes*, we try to leverage the concept of structural metrics from software engineering, as e.g. used in metrics catalogs for object-oriented designs, and use it for evaluating application landscapes.

1.3. Scope and approach of the *Enterprise Architecture Management Tool Survey*

The understanding of what EA management is about and what it is not about varies widely. In Section 3.1, we will introduce our understanding and the understanding of our partners, which will be used in this survey. Due to the fact, that the application landscape plays a prominent role in the context of EA management, the survey is focused on the support application systems provide for the business. Thereby, different views on the application landscape need to be considered to address the various concerns of multiple stakeholders, which need to work collaboratively on the evolution of the EA.

What this survey is specifically not(!) about is *business process modeling and management*, *systems management*, *testing (functional, coverage, regression, load)*, etc. In general, this survey deals, among others, with *demand management, IT architecture management*, *application management*, *infrastructure management*, *multiproject and synchronization management*, *project portfolio management*, and *strategies management*. In the context of EA management the above areas have to be considered with a high level of abstraction. Project management in EA management means, that information about projects is linked to other information objects of the EA (e.g. linking projects with application systems). Details about e. g. people, resources, and activities are likely to be not of relevance.

Other parts of the survey deal with criteria about architecture management, impact analysis, and strategy management. Thereby, exemplary questions an EA management tool should be able to support answering:

- What does my business and IT architecture look like? How will the application landscape look in the future?
- What happens if an object is changed? Which other objects in the EA are affected?
- Which action items and goals are derived from which strategies and how do they affect the EA?

The approach of EAMTS2008 is based on a list of criteria and a set of scenarios to analyze the capabilities of the EA management tools to answer the questions as listed above. An initial draft of the list of criteria and the set of scenarios was built on the wide experience gathered during the software cartography project and the conduction of the *Enterprise Architecture Management Tool Survey 2005*. These initial drafts where subsequently rewritten, refined, consolidated, and enhanced by the inputs from and insights of our sponsors and partners during three extensive workshops.

The list of criteria consists of a comprehensive catalog of questions concerning, among others, tool data (e. g. release information, history), tool architecture (e. g. supported platforms, infrastructure requirements), and collaboration support (e. g. locking, collaborative work). The complete list of criteria is contained in Appendix A. The tool vendors answered the list of criteria in order to provide additional information to the simulation results. The answers of the different tool vendors are not fully printed in this survey, but can be requested from *sebis* or from the tool vendors that participated in the survey.

Besides evaluating the tools against our list of criteria, we validate the existence of both the specific functionality of the tool and the support the tools offer for EA management tasks by simulating a set of scenarios. The scenarios are organized into two groups. The scenarios in Section 4.1 are analyzing specific functionality of the tools, e. g. the creation of visualization, the support provided for multiple users and collaborative works, or the adaptability of the information model. The second group of scenarios in Section 4.2 analyzes the support the tools offer for specific EA management tasks and processes, like application landscape management, project portfolio management, or application architecture management. Each scenario is described using the concerns relevant to EA management, which are addressed by the scenario, the questions derived from these concerns, the tasks, which are executed during scenario simulation, and the deliverables, which should be created in the scenario simulation.

The evaluation results are not compiled into a simple ranking of the tools, but are aggregated by two kiviat diagrams. The first diagram reveals how specific functionalities are implemented in the respective tool, while the second diagram focuses on the strength and weaknesses an user is likely to face, when carrying out each of the EA management tasks as exemplified in the survey. This fine-grained result enables the reader to individually match their specific requirements with the tools evaluated.

1.4. Structure of the *Enterprise Architecture Management Tool Survey*

In order to reflect the requirements on the tools, which according to the sponsors and partners of the survey should be analyzed, we decided to pursue a threefold evaluation approach, relying on two distinct sets of scenarios together with an online questionnaire – the list of criteria – each vendor had to answer. The first set of scenarios focuses on specific functionality, an EA management tool should provide, without connecting these functionalities to the execution of a typical EA management task. These tasks are subsequently evaluated by the scenarios of the second group, which reflect tasks, that have in cooperation with the sponsors and partners been identified as essential constituents of many EA management endeavors. For ensuring consistency and continuity of the simulation, an exemplary set of data describing the enterprise architecture of a fictitious department store, so-called *SoCaStore* has been developed by us. This data and the underlying information model used therefore is subsequently introduced in Chapter 4, preceded by a short description of the EA management approach pursued throughout the survey.

Chapter 4 introduces the two sets of scenarios as used for evaluating the EA management tools. The first set, intented for analyzing specific functionalities of the tools, constitutes of the following scenarios

- Importing, Editing, and Validating Model Data (see Section 4.1.1),
- Creating Visualizations of the Application Landscape (see Section 4.1.2),
- Interacting with and Editing of Visualizations of the Application Landscape (see Section 4.1.4),
- Supporting lightweight Access (see Section 4.1.5),
- Editing Model Data using an external Editor (see Section 4.1.6),
- Adapting the Information Model (see Section 4.1.7),
- Handling large scale Application Landscapes (see Section 4.1.8), and
- Supporting multiple Users and collaborative Work (see Section 4.1.9).

A final Section 4.1.10 documents, how the scenario simulation for the above scenarios is conducted.

Each of the scenarios for analyzing, how the tool supports a typical EA management task, is organized in the same way, first giving a short motivation for the specific task, then outlining the objectives to be achieved, and concluding with a set of exemplary deliverables. The scenarios in the second set are:

- Landscape Management (see Section 4.2.1),
- Demand Management (see Section 4.2.2),
- Project Portfolio Management (see Section 4.2.3),
- Synchronization Management (see Section 4.2.4),
- Strategies and Goals Management (see Section 4.2.5),

- Business Object Management (see Section 4.2.6),
- SOA Transformation (see Section 4.2.7),
- IT Architecture Management (see Section 4.2.8), and
- Infrastructure Management (see Section 4.2.9)

A final Section 4.2.10 describes, how the scenarios above are simulated with each tool.

1.5. Analyzed EA Management Tools

At the beginning of this tool survey, we compiled a list of vendors positioning their tools in the field of EA management (see Table B.1). Having compiled this list, we began a discussion with our industry partners, in order to concentrate our efforts on those tools that were of major interest. The shortened list containing the tools, which have been selected by our sponsors and partners for evaluation (with the denoted versions) is shown in Table 1.1.

No	Name of Vendor	Name of Tool(s)	Version
1	Adaptive	Adaptive EAM	5.0
2	alfabet AG	planningIT	3.1
3	BOC	ADOit	3.0
4	Casewise	Corporate Modeler Suite, IT Architecture Accelerator[a]	
5	Embarcadero	EA/Studio[b]	1.1
6	Hewlett Packard	Mercury Project and Portfolio Management Center[c]	
7	IBM	Rational Software Architect[d]	
8	IDS Scheer AG	ARIS IT Architect	7.0.2
9	IDS Scheer AG	ARIS ArchiMate Modeler[e]	
10	MEGA International SA	MEGA Modeling Suite 2007	2007
11	Metastorm	ProVision	6.0
12	Telelogic AB	Telelogic System Architect	11.0
13	Troux Technologies Inc.	Troux	7.0

[a] See 8 for a statement why this tool was not evaluated.
[b] Embarcadero EA/Studio was chosen in accordance with our sponsors and partners to include a tool, which has newly entered the EA management market.
[c] See 10.1 for a statement why this tool was not evaluated in detail.
[d] See 11.1 for a statement why this tool was not evaluated in detail.
[e] See 12.4 for a statement why this tool was not evaluated in detail.

Table 1.1.: Analyzed EA Management Tools

CHAPTER 2

Executive Summary

Contents

This chapter gives a brief summary of the results of the *Enterprise Architecture Management Tool Survey 2008.* In order to provide more than a simple ranking of the tools, kiviat diagrams[1] are used, which support the comparing and rating of the tools evaluated according to specific functionalities and the EA management support provided. These kiviat diagrams are subsequently introduced in Section 2.1 and latter used to give an overview on the results of the tool evaluation in Section 2.2, which is complemented by a short textual description for every tool evaluated. Detailed information about the results is provided in an additional chapter for each of the tools.

[1]Kiviat diagrams are also known as spider diagrams.

2.1. Introducing Kiviat Diagrams for Evaluation

Our evaluation focuses on different aspects of the tools' overall performance. Reflecting this, we developed two kiviat diagrams, each concerned with particular aspects of the tool's capabilities. Section 2.1.1 introduces the kiviat diagram, which evaluates the tool's specific functionalities like the flexibility of the information model, the capabilities to create visualizations, usability, etc. The second kiviat diagram deals with EA management support, e.g. landscape management, SOA transformation and is introduced in Section 2.1.2. In both diagrams we use an ordering reflected by a rating between 1 (low capability) and 7 (high capability) for each axis[2]. Thereby, it has to be noted, that the score along the individual axes is a result of an ordering of the assessed solutions. Comparison of the rating between different axes is therefore not meaningful.

2.1.1. Kiviat Axes for Specific Functionality

The kiviat axes described in this section (see Figure 2.1) are derived from the main concerns addressed in the list of criteria (see Chapter A) and the simulation of the scenarios for analyzing specific functionality (see Section 4.1). The mapping of the different scenarios and questions from the list of criteria to the axes is outlined in the following.

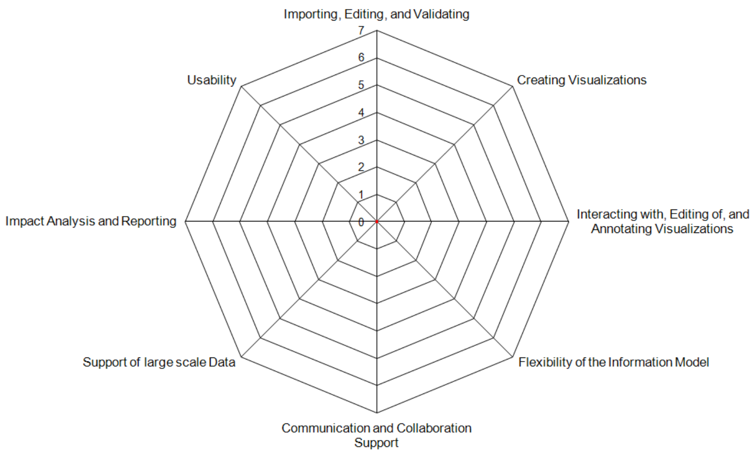


Figure 2.1.: Kiviat Diagram visualizing specific Tool Functionality (see text)

The axis *Importing, Editing, and Validating* reflects the tool's capabilities of importing and exporting data in different formats. In addition, the provided capabilities to validate the imported data during the importing process are evaluated as well as the editing mechanisms provided. The importing capabilities are particularly important, as companies often have gathered information prior to introducing an EA management tool. The functional part of importing regards, if any entity type and relationships can be imported or if importing is limited to just a few types and relationships, and in addition, if all attributes of the entity or only a subset of them can be imported. The validation criteria is concerned with the handling of e. g. data type consistency checks or attributes, which are marked as mandatory. Furthermore, the tool's capabilities to provide editing functionalities are evaluated. Thereby, the editing of multiple data or the usage of pick-lists, which provides a convenient way of handling the tool, as well as supported external editors are examined.

[2]The predecssor of this report [se05] used a rating between 1 (low capability) and 5 (high capability), this rating was extended to provide a more detailed analysis of the performance of the tools evaluated.

The axis *Creating Visualizations* focuses on the presentation capabilities offered by the tool. The functionalities evaluated in this category contain common forms of visualization techniques and visualization types such as *portfolio matrices* and *software maps* (see Appendix C). The tool's built-in capabilities to visualize specific relationships between entities, e.g. a *part-of* relationship by using containment or a vertical/horizontal alignment of symbols also contributes to the evaluation result for this axis. Furthermore, the axis refers to the tool's capabilities of defining new visualization templates. In this context *template* means *kind of visualization*, not instance - thus defining a new template creates a new kind of diagram with rules defining the entities to be displayed and manners how they shall be displayed. This templating does not necessarily contain capabilities of performing automated layout, although this function would contribute to a positive ranking.

The axis *Interacting with, Editing of, and Annotating Visualizations* deals with the tool's capabilities to support the handling of visualizations. The interacting capabilities are particularly important, as visualizations are often used to illustrate complex interrelations, which ask for interacting possibilities like zooming in and out of visualizations, layering, or the closing of symbols to hide a complex inner structure. Furthermore, the provided possibilities to edit visualizations contribute to the evaluation results of this axis. Thereby, a positive impact on the ranking is reached, if the tool supports manual adaptations of visualizations and if it further distinguishes between adaptations, which do not or do change the visualization's semantics. In addition, this axis refers to the tool's capabilities to annotate visualizations with certain information. These annotation mechanisms are particularly important to further characterize elements of the application landscape utilizing attribute values or calculated indicators (e. g. metrics). Thereby, the possibilities provided and the number of annotations supported for a symbol are evaluated.

The axis *Communication and Collaboration Support* refers to the collaboration support provided by the tool, whether concurrent working on the same data is possible or not and which limitations to concurrent work exist. In the context of collaborative work, the auditing mechanisms provided by the tool are of particular importance, as well as workflow or notification capabilities. It is of interest, if role or user based access control is possible, and on which levels of granularity it can be configured, e.g. on model-level, entity-level or attribute-level, and which kinds of access rights can be granted. The availability of a web front end with limited or limitable editing capabilities as a lightweight input interface or reader also contributes to a positive ranking in this category, as well as the availability of clients for offline working, which means working without having access to the repository. In this context, the tool's versioning capabilities for collaborative work on model element level are evaluated, concentrating on features, which help the users to merge independently edited models or parts of them, allowing to resolve conflicts automatically, if possible.

The axis *Flexibility of the Information Model* reflects the requirement that the information model of the EA management tool should stay adaptable, even if data has already been imported into the tool. We evaluate the possibilities to create, edit, and delete entities, attributes, and relationships of the predefined information model in order to match the concepts of an enterprise-specific information model. In order to perform a mapping between the predefined information model and the enterprise-specific one it is also of interest, if the information model can be visualized or exported and (re-)imported. Furthermore we evaluate, if it is possible to supply default values for attributes or to supply constraints to selected concepts, e. g. flag attributes as *mandatory*. Also the richness of the type system supplied is taken into consideration. How these configurations can be performed is also reflected in this axis' ranking, e.g. if a graphical user interface for performing these changes is available or if a kind of scripting language has to be used. Additionally, we consider if the *customer* is provided with the needed functionality to adapt the information model to individual needs, or if only the vendor is capable of changing the information model, which leads to a lower ranking.

The axis *Support of large scale Data* refers to the tool's capabilities to handle large scale application landscapes containing thousands of application systems and interconnections between them. We evaluate the tool's capabilities to import, edit, and visualize large amounts of data as they might exist in large enterprises. On the one hand, aspects of performance are considered and influence the ranking of the tool. On the other hand, the convenient tool usage is evaluated as e. g. smart support for picking an element from a list of several hundred entries or aggregated visualizations, which lead to a higher ranking, if provided.

The axis *Impact Analysis and Reporting* mirrors the ability of the tool to perform calculations or impact analysis based on the data contained in the repository. Calculations in this specific case refer e. g. to summing up or deriving averages of values of an entity's attributes as well as of a transitively linked entity. Impact analyzes employ traversing specific relations and filtering the resulting data according to given criteria. It is especially of interest, whether these capabilities are built into the tool's graphical user interface, if they are reachable via a query language, or if they have to be added by the user programmatically. The axis is further concerned with the tool's capabilities of creating tabular reports of the repository data. There are several dimensions of configurability which are of interest concerning this ability. First, configurability of the selection of entities reported, whether there is a way to filter the entities by particular criteria or not. These filtering criteria might be of a certain complexity including boolean expressions on values of standard and user-defined attributes. Second, the configurability of the projection of the attributes reported is of interest. Although it is of a certain interest, if the tool can report all attributes of an entity type including user defined attributes in a convenient manner, it is also important, that this set of attributes can be customized in order to create specially adaptated reports.

The axis *Usability* refers to the general user experience. It does not reflect, if a specific functional task could be accomplished, but it reflects how simple or complicated it was to use the provided functionality. Qualities in regard to this axis could be, but are not limited to, intuitive and well structured menus with articulative names, consistent property editors, that allow editing model data without requiring many different editors or editing dialogs, and a supportive, well indexed help system with a continuous wording and clear examples. This axis reflects the subjective impression resulting from the tool analyses and the scenario simulation. Especially pitfalls experienced therein are considered in this ranking.

2.1.2. Kiviat Axes for EA Management Support

The kiviat axes described in this section (see Figure 2.2) correspond one-to-one to the scenarios in Section 4.2. The comparison of the simulation of theses scenarios leads to the rating of the tools on the corresponding axes.

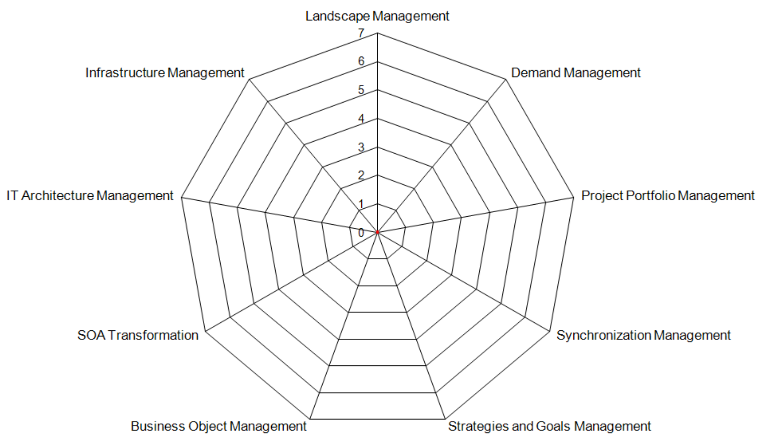


Figure 2.2.: Kiviat Diagram visualizing Tool Support for EA management Tasks (see text)

The axis *Landscape Management* considers the tools support for future planning and development of the application landscape. Thus, the tools capabilities to create different planned landscapes based on the current landscape as well as a target landscape is evaluated. Concerning aspects of the evolution of the application landscape, the support of the tool for two dimensions of time – planned *for* and planned *at* a certain time – as well as the possibilities to create variants of a planned landscape for a specific time are evaluated. Thereby, it is also taken into consideration, if the different variants of the planned application landscape could be derived from the respective project portfolios. The possibility to historicize current, planned, and target landscapes in the tool to enable comparisons leads to a higher ranking, if provided.

The *Demand Management* axis reflects the tool's capabilities to gather, document, and process demands originating from business and IT. Thereby, the tool's capabilities to store demands and link them to the affected elements of the EA are of particular importance to identify similar demands and map them to project proposals. To prepare the project portfolio management the tool's capabilities to identify demands asking for similar functionality or affecting the same application system are evaluated. Furthermore, the possibility to transform a demand seamlessly into a project proposal and therefore reuse information already stored in the tool is evaluated.

The *Project Portfolio Management* axis covers the management process concerned with deciding on the project portfolio for the next planning period. Thereby, projects on an EA management level of abstraction, including their proposal, selection, approval or cancellation are considered. On the one hand, the evaluation focused on the capabilities of the tool to link the project proposals to affected elements of the EA, thus being connected with several artifacts like business processes, business objects, application systems, architectures, etc. On the other hand, the provided possibilities to conduct different kinds of analyzes on the portfolio, e. g. cost calculations, are evaluated.

The axis *Synchronization Management* mirrors the tool's capabilities to address issues of synchronizing projects, according to their interdependencies derived from the objects (e. g. application systems or services) a given project is likely to change. In this case, two projects that modify the same application system at the same time may produce a conflict that is deemed worth avoiding. Therefore, the tool's capabilities to support multiproject management are considered. Of course, multiproject management in this context has to be seen at an EA management level. Project dependencies between new and already running projects have to be coordinated and their relationships to other artifacts of the EA neet to be identified and managed. This is one of the major activities, when projects changing the same artifact of the EA have to be re-scheduled due to the delay of one project. Therefore, the axis is called Synchronization Management.

The axis *Strategies and Goals Management* covers issues of aligning the EA management activities to the organization's strategies and goals. It is analyzed, whether the tool supports operationalizing a strategy through the organizational hierarchy by decomposing it into smaller and more detailed pieces. This also includes tracing back the decomposition process, making it possible to trace from a specific action item on a fine grained level of the organizational hierarchy to the strategy it has been derived from. Furthermore, the provided possibilities to control the fulfillment of the goals defined, like *Balanced Score Cards*, are evaluated. Those functionalities may cover different artifacts of EA management, not limited to projects, and should be available in an integrated manner within the EA management tool.

The *Business Object Management* axis refers to the tools capabilities to manage business objects as well as the operations performed on them, and the exchange of the business objects between application systems to support business processes. A business object is a representation of an artifact or concept of relevance for business. They are characterized by type, name, description, and may have attribute values, status and relationships. This axis reflects the tool's capabilities to provide an overview about the business objects involved and exchanged during the execution of a business process. Thereby, functionalities to e. g. identify business processes that need manual operations during their execution or application systems that hold the master copy of a business object have a positive influence on the ranking of the tool, if provided.

The *SOA Transformation* axis reflects the tool's capabilities to support the enterprise in transforming their architecture into a *service oriented architecture* (SOA). The capabilities provided by the tool to identify candidates for reusable services are evaluated leveraging a top-down as well as a bottom-up approach. The top-down approach identifies services according to the usage of business objects within the conduction of different business processes, whereas the bottom-up approach identifies technical functionalities currently provided by applications, which should be transformed into reusable services to demonstrate the benefit of the transformation. Besides the functionalities provided by the tool to identify applicable candidates for the transformation into a service, the effects the transformation will have on the application landscape should be visualized, which leads to a higher ranking, if provided.

The axis *IT Architecture Management* deals with the introduction and the implementation of blueprints standardizing the architecture of specific application systems. This means introducing architectural standards future application systems should be based on and current application systems should be adaptated to. The concept of an architectural blueprint combining abstract technologies and architectural solutions integrating technologies is evaluated. As an example a *4-tier-thin-client-architectural blueprint* may be made up of a thin client, a web server, an application server, and a database management system; a corresponding architectural solution would realize the blueprint using, e.g. an Apache HTTP-Server as a web server, a BEA WebLogic Server as an application server, etc. The goal of this axis is to evaluate the provided tool support during the consolidation and homogenization of the used architectures in the application landscape.

The axis *Infrastructure Management* is concerned with managing infrastructure elements, such as middleware and hardware systems, storage services, etc. used by application systems and services. Infrastructure management is about optimizing IT operations ensuring infrastructure service quality, and reducing operating as well as maintenance costs. This may be achieved e.g by consolidating infrastructure elements for homogenization purposes and consolidating resources for cost reduction. Thereby, the tool may help to identify a call for action concerning infrastructure elements. Furthermore, the provided functionalities of the tools to identify other elements of the EA like application systems, organizational units, or business processes, which are affected by the proposed change, are evaluated.

2.2. Tool Evaluation at a Glance

The following sections contain aggregated and condensed results of the tool evaluation. For each tool, the two kiviat diagrams *specific tool functionality* and *EA management tasks* show the results of our scenario simulation and information provided by the tool vendors according to the list of criteria. Additionally, each section contains a short abstract for each tool giving a brief introduction to the approach the tool takes to support EA management.

For each axis of the kiviat diagrams, the evaluation results in a value from 1 to 7. The values are ordinally scaled. Transfered to the kiviat axes used for evaluation, this means that a tool rated 6 on one axis is better than a tool rated 5, but this does not provide information to derive *how much better* the 6 rated tool is. Finally, we want to emphasize the kiviat diagrams only being the high level results of the survey, for a detailed evaluation we refer to Sections 5 to 16.

2.2.1. Evaluation of *Adaptive EAM* from Adaptive Inc.

Adaptive EAM pursues an integration approach to EA management in a threefold way. First, via its strong repository hosting a predefined but highly adaptable information model, integrating a multitude of modeling facilities, as the UML or SPEM. Second, via the tool's architecture especially designed for aggregating and integrating information drawn from various sources. Finally, by the strong commitment to industry standards, as the SQL for querying, which both fosters user familiarization and leverages possibilities to integrate with third party tools.

When executing EA management, the flexibility of the tool allows the user, to easily create reports and queries, which are useful in different typical EA management tasks, e. g. *Business Object Management*. The integration approach is consequently complemented by an adaptable and configurable web-based user interface, providing rich functionalities for collaborative work of a distributed user group.

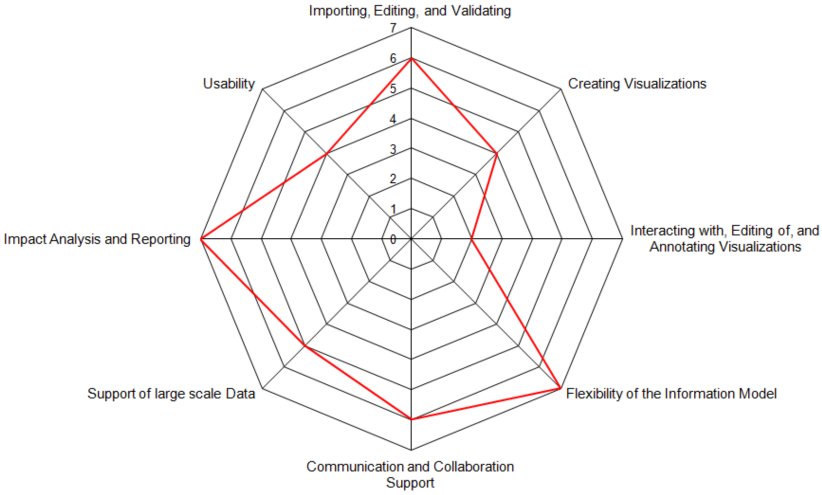


Figure 2.3.: *Adaptive EAM*: Kiviat Diagram for specific Tool Functionality

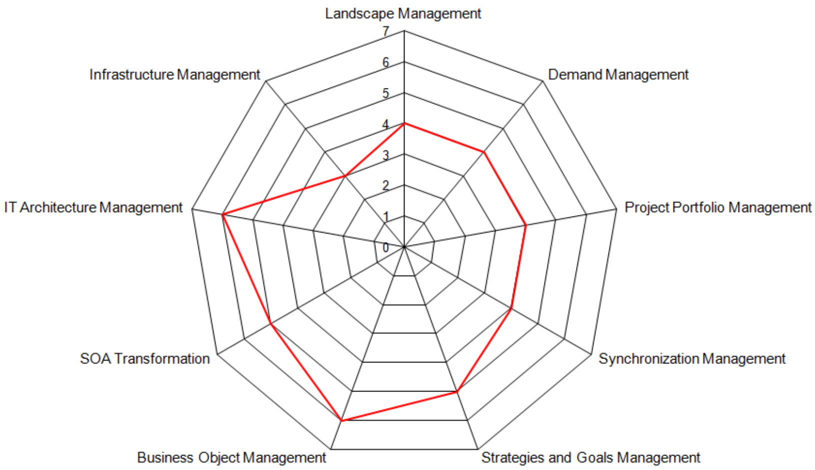


Figure 2.4.: *Adaptive EAM*: Kiviat Diagram for EA management Tasks

2.2.2. Evaluation of *planningIT* from alfabet AG

planningIT pursues a process driven approach to support every typical activity in EA management. Thereby, the tool provides a repository with a given information model, which builds the foundation for the numerous functionalities provided by the tool. These out-of-the-box functionalities include e. g. predefined reports and diagram types, which can be created automatically, as well as an easy to use *query builder* providing strong capabilities to perform customized analyses on the governed data.

As alfabet does not regard EA management as an endeavor performed by a single person, *planningIT* provides various functionalities to support collaborative work, e. g. a web-based thin client, role-based access, and workflow mechanisms. These capabilities can be leveraged to support the users during the conduction of everyday EA management tasks, e. g. *landscape management*. Further, the user is supported by the integrated planning process and the given methodology provided by *planningIT*.

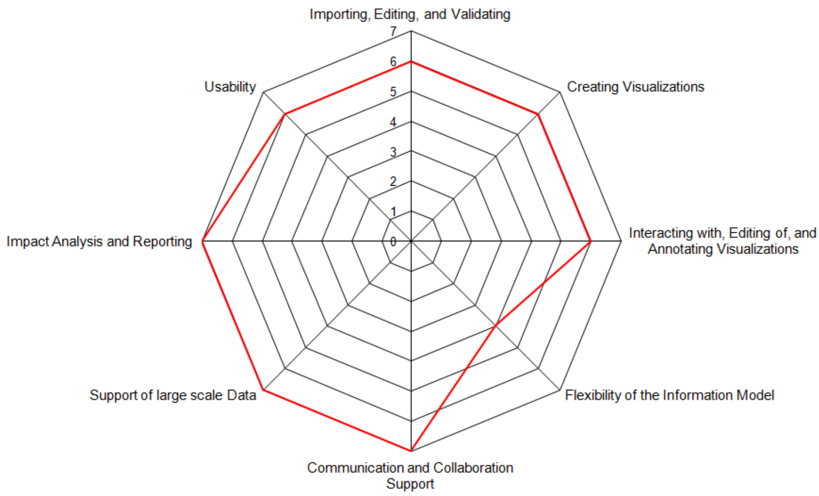


Figure 2.5.: *planningIT*: Kiviat Diagram for specific Tool Functionality

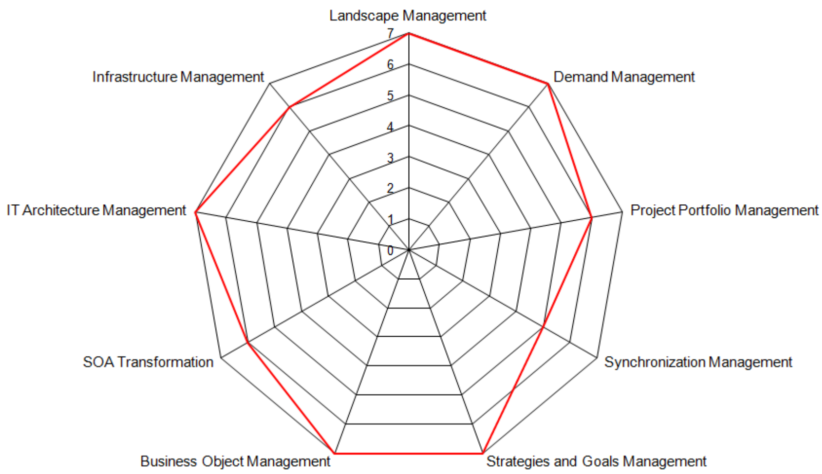


Figure 2.6.: *planningIT*: Kiviat Diagram for EA management Tasks

2.2.3. Evaluation of *ADOit* from BOC GmbH

ADOit pursues a metamodel driven approach to be fully configurable to the requirements of the customer. The preconfigured information model is organized in different layers, e. g. business architecture, application architecture, and software architecture, of which each one offers predefined diagram types and classes. The information model can be extended and enhanced by BOC consultants or via a convenient graphical user interface. Additional functionality like a view generator and a query language is provided, which can be further adapted according to customer-specific needs.

ADOit does not see itself as a tool that is meant to cover all EA management aspects at once. Instead it provides means to implement EA management in a step-by-step approach addressing typical tasks, e. g. *landscape management*. Having performed a customization to the tool, an everyday user is able to perform the various tasks via an easy to use and highly adaptable user interface.

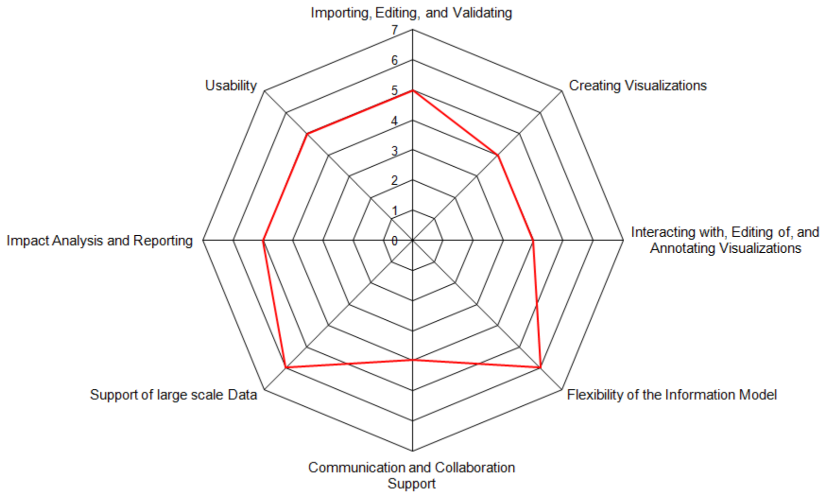


Figure 2.7.: *ADOit*: Kiviat Diagram for specific Tool Functionality

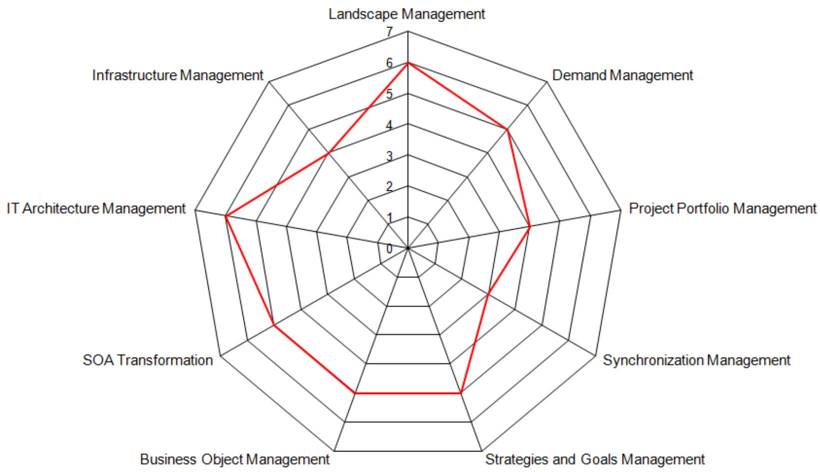


Figure 2.8.: *ADOit*: Kiviat Diagram for EA management Tasks

2.2.4. Evaluation of *Embarcadero EA/Studio* from Embarcadero Technologies Inc.

Embarcadero EA/Studio comes shipped with a lightweight predefined information model, of which the focal point lies on *data* and *business process modeling*. Consequently, the predefined visualization techniques are especially designed for graphical modeling in these domains, complemented by capabilities, for performing analyses on the respective concepts, e. g. business processes. The set of functionalities can be accessed via a well-structured user interface, built on the eclipse rich client platform.

Due to the limitations of the information model of *Embarcadero EA/Studio*, we have decided in accordance with the team of Embarcadero Technologies Inc. not to evaluate the tool in respect to the support for EA management tasks, as such evaluation would have created no sensible result. According to the vendor, it is planned to extend the predefined information model in future release, such that the functionalities described in Section 9.1 could be leveraged in the context of EA management.

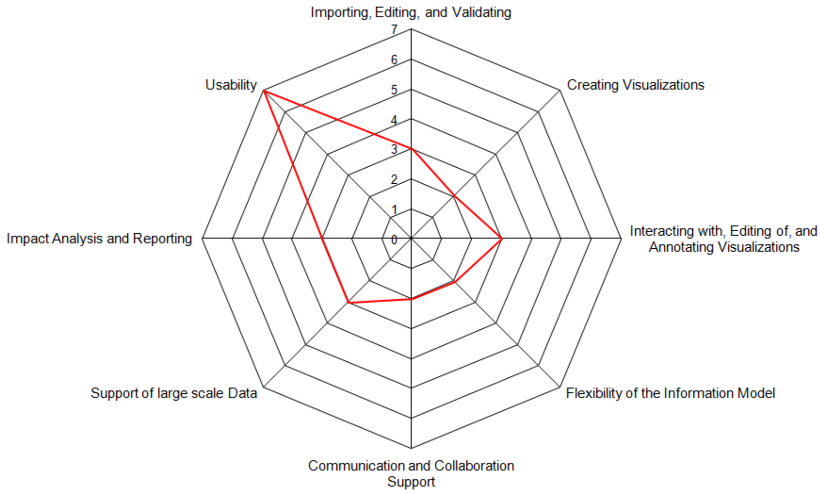


Figure 2.9.: *Embarcadero EA/Studio*: Kiviat Diagram for specific Tool Functionality

2.2.5. Evaluation of *ARIS Platform* from IDS Scheer AG

The *ARIS Platform* pursues a strongly methodological approach to EA management, backed by a comprehensive information model, spanning concepts from business to infrastructure level. In addition, the methodology defines a set of predefined visualization techniques, allowing the user to focus on specific aspects of EA management, typically related to a specific management task. The visualizations are further complemented by a set of standard reports, which can be accessed in many different formats.

The adoption of industry standard methodologies, as the balanced scorecard for *strategies and goals management*, strongly facilitates user familiarization. Besides this, a number of manuals and user guides, foremost the method manual, provide in-depth information on the *ARIS* approach to EA management.

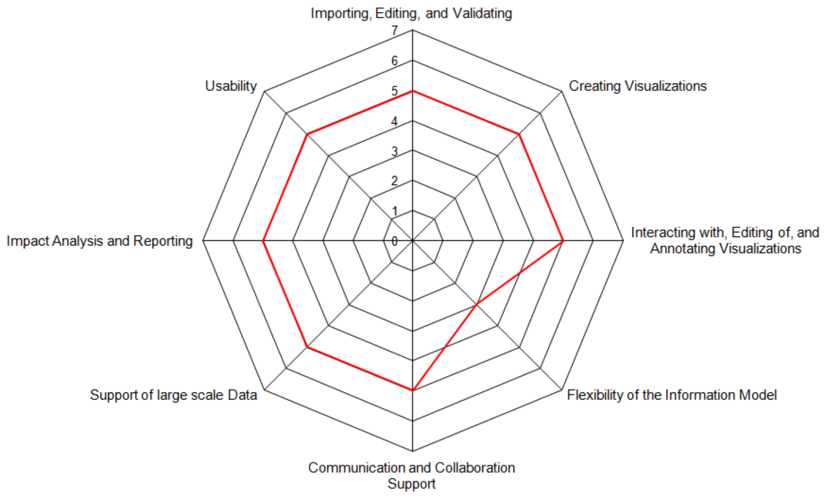


Figure 2.10.: ARIS Platform: Kiviat Diagram for specific Tool Functionality

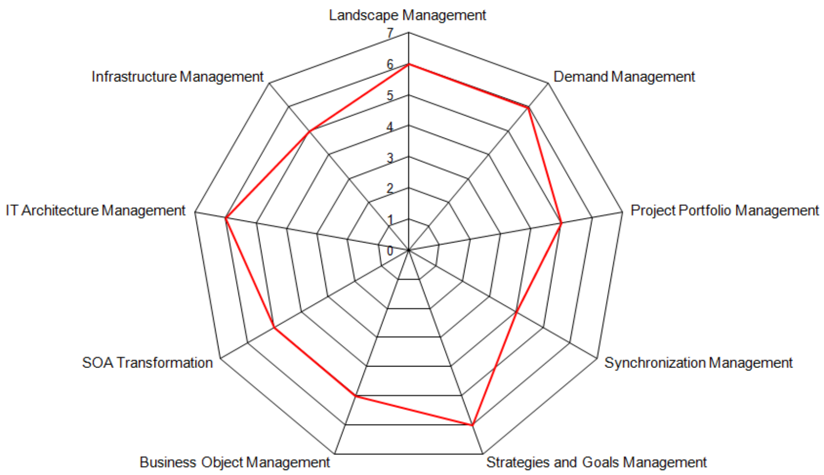


Figure 2.11.: ARIS Platform: Kiviat Diagram for EA management Tasks

2.2.6. Evaluation of *MEGA Modeling Suite* from MEGA International SA

MEGA Modeling Suite's approach to EA management has an emphasis on metamodeling, complemented by a rich set of built-in analyses and reports based on the compulsory predefined information model. The analyses can therein be executed based on user defined queries against information stored in the tool. In order to take full advantage from previously created visualizations, the analysis capabilities provide the user with an outstanding, highly configurable visual annotation mechanism, which can graphically display the results of an analysis on any visualization containing an affected element.

Based on this and a multitude of other capabilities, the user can easily execute typical EA management tasks - especially taking full advantage of the tool's functionalities in *Synchromization* and *Infrastructure Management*. Particularly in the execution of the first task *MEGA Modeling Suite*'s deep understanding for temporal project dependencies has to be noted.

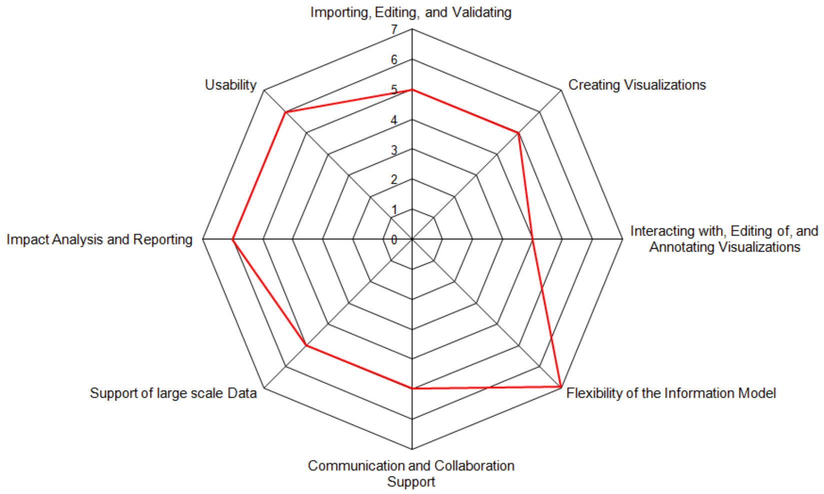


Figure 2.12.: *MEGA Modeling Suite*: Kiviat Diagram for specific Tool Functionality

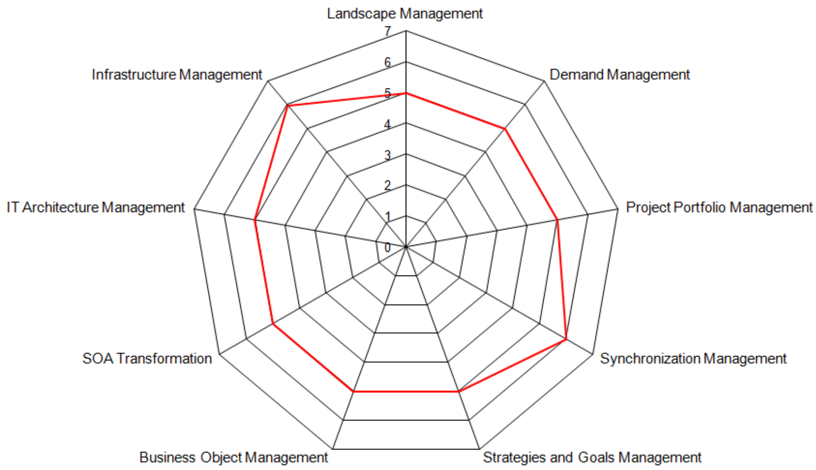


Figure 2.13.: *MEGA Modeling Suite*: Kiviat Diagram for EA management Tasks

2.2.7. Evaluation of *ProVision* from Metastorm

Metastorm ProVision pursues a metamodeling approach to EA management with a focal point on modeling. Thereby, the approach centers on an easy to learn and easy to use graphical interface for editing model data. The predefined information model of *ProVision* supports different frameworks, e. g. *DoDAF* or *Zachmann* and can in addition be easily customized.

The flexibility of the information model and the shipped visualizations offers the possibility to define and roll out an enterprise-specific methodology for the EA management. In order to perform the typical EA management tasks, e. g. *demand management, strategies and goals management* or *business object management*, predefined concepts, reports, and visualizations can be used or customized according to the specific needs. Further, an everyday user is likely to take advantage from the well-structured and intuitively understandable interface provided by *ProVision*.

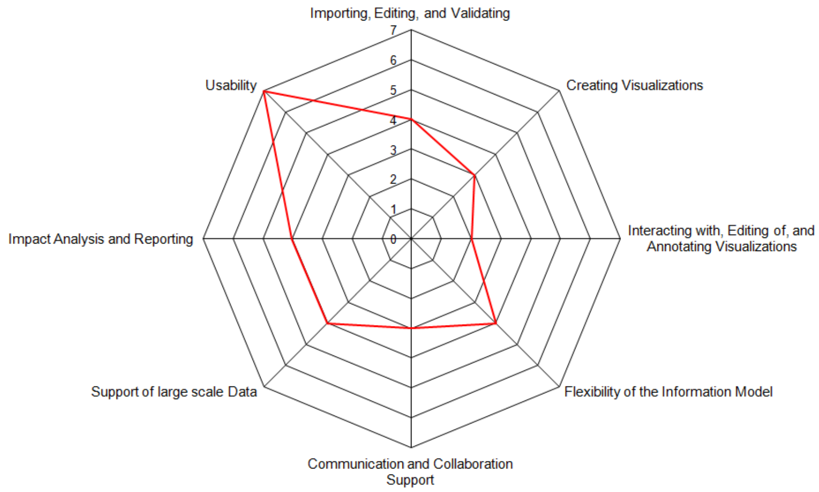


Figure 2.14.: *ProVision*: Kiviat Diagram for specific Tool Functionality

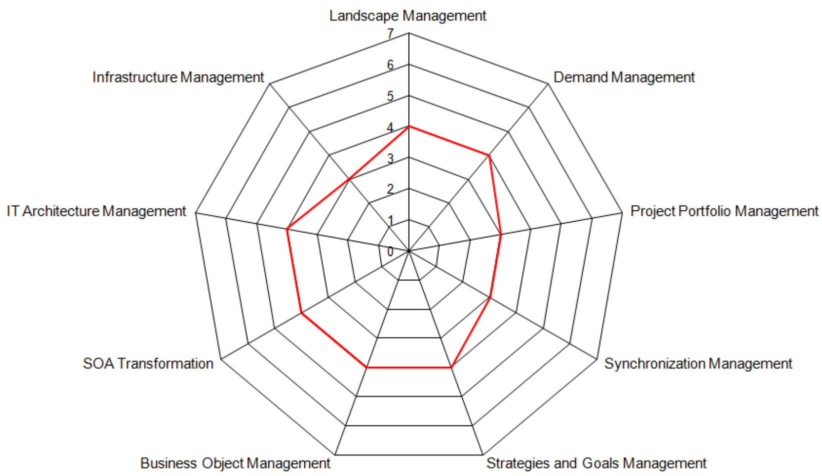


Figure 2.15.: *ProVision*: Kiviat Diagram for EA management Tasks

2.2.8. Evaluation of *System Architect* from Telelogic AB

The *System Architect* pursues a metamodel oriented approach to allow adaptations according to the specific needs of a customer. The information model adaptations can be complemented by changes to available and new diagrams and symbols used for visualizing different aspects of the EA. Thereby, the user can leverage the rich visualization possibilities provided by *System Architect* complemented by the possibility to automate their creation using the integrated scripting capabilities.

System Architect provides flexible and strong functionalities for performing analyses on the governed data by the integrated SQL-like query language. Thereby, this query language can be used to create graphical as well as textual reports, enabling the conduction of EA management tasks, e. g. *strategies and goals management*. Furthermore, an everyday user is supported by an easy to use graphical user interface, which can additionally be customized.

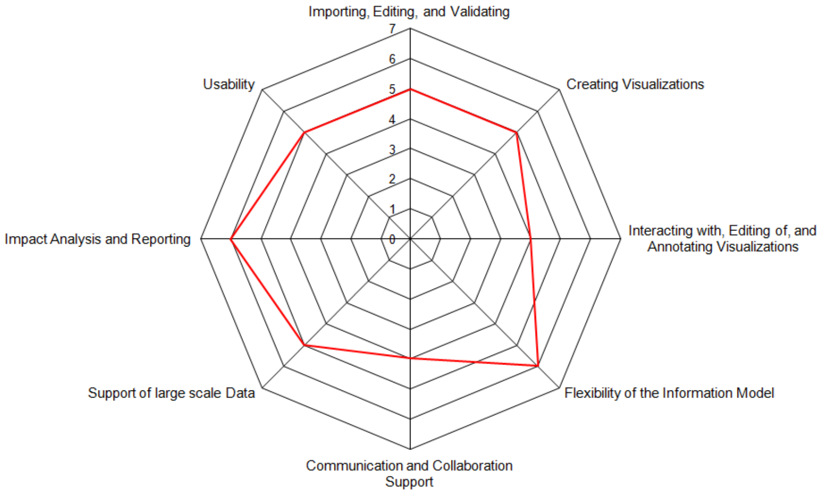


Figure 2.16.: *System Architect*: Kiviat Diagram for specific Tool Functionality

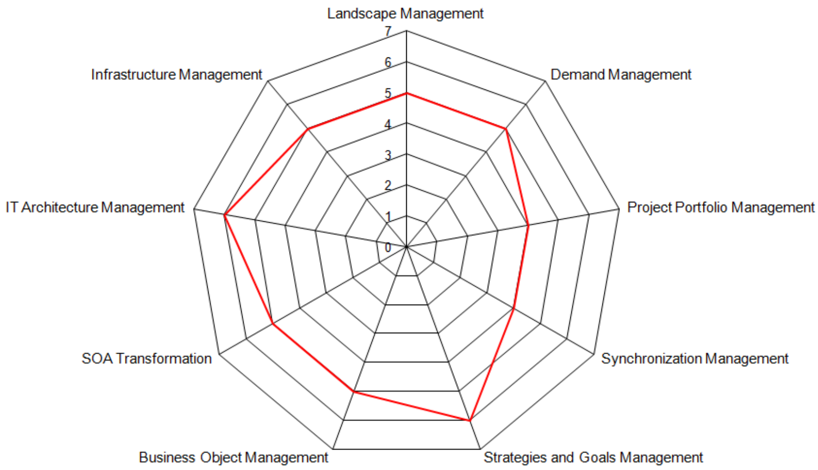


Figure 2.17.: *System Architect*: Kiviat Diagram for EA management Tasks

2.2.9. Evaluation of *Troux* from Troux Technologies

The *Troux 7* pursues a metamodel oriented approach to allow customization according to the specific needs of the customer. To facilitate the execution of EA management tasks in the context of strategic IT planning, *Troux 7* provides strong capabilities for performing analyses on the governed data via the integrated *Troux Intelligence*. In addition the user can leverage the flexibility of visualizations complemented by the possibility to automate their creation by customization.

As *Troux* does not regard EA management to be a management endeavor governed by a single tool, the *Troux Metaverse* provides comprehensive functionalities to gather and integrate information from various sources. This functionality is useful during EA management tasks, e. g. *project portfolio management* or *SOA transformation*. An everyday user can take advantage of the rich functionalities as the user interface of *Troux 7* is well-structured and easily understandable.

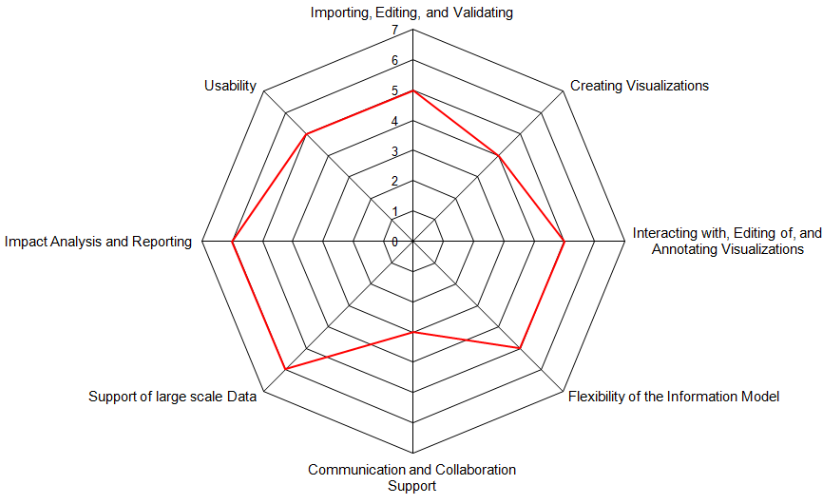


Figure 2.18.: *Troux 7*: Kiviat Diagram for specific Tool Functionality

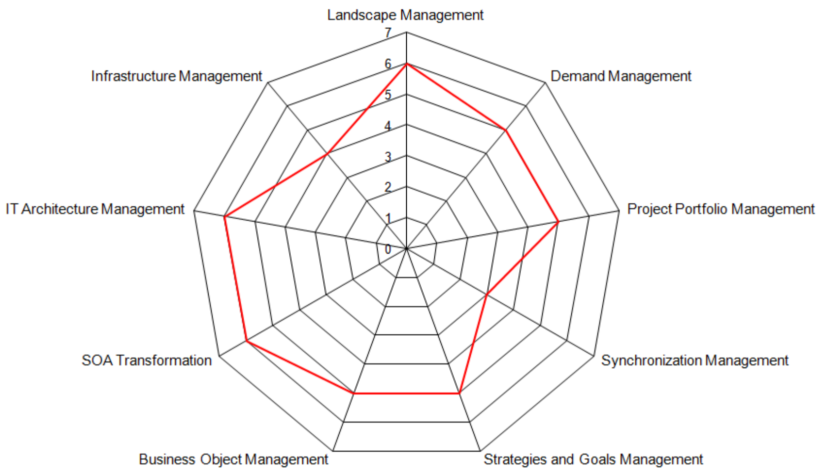


Figure 2.19.: *Troux 7*: Kiviat Diagram for EA management Tasks

CHAPTER 3

Approach to Enterprise Architecture Management

Contents

The *Enterprise Architecture Management Tool Survey 2008* is based on the knowledge we gained in the last years during our research project *software cartography* and it's successor research project *system cartography*[1]. The results of our research are strongly influenced by the information, insights, and discussions we gathered in the cooperation with our industry partners. The following sections provide an introduction to our EA management approach (cf. Section 3.1) consisting of starting points for EA management requirements and tasks, the need for a holistic, integrated information model, the interaction of EA management and related management processes, and an approach to quantify aspects of the EA. Finally, Section 3.2 presents and explains the information model used for simulating the scenarios.

3.1. Introduction to EA Management

Although EA management has been gaining importance in research and practice during the last years no common definition for EA management exists. Instead, a variety of definitions by many different

[1]For further details on the research project system cartography, please visit www.systemcartography.info

organizations are available. In order to support a common understanding of EA management and a consistent evaluation of tools in this survey, we subsequently provide the definition of EA Management, which is used throughout the remainder of the survey.

> *Enterprise architecture management* is a continuous and iterative process controlling and improving the existing and planned IT support for an organization. The process not only considers the information technology (IT) of the enterprise, also business processes, business goals, strategies etc. are considered in order to build a holistic and integrated view on the enterprise.
>
> Goal is a common vision regarding the status quo of business and IT as well as of opportunities and problems arising from theses fields, used as a basis for a continually aligned steering of IT and business.

EA management is the discipline of planning and managing the evolution of the EA. On the one hand, EA management should be driven by and adjusted to the business strategy and on the other hand, IT should play the role of an enabler for the business strategies and goals: One of the main efforts of EA management is the *alignment of business and IT*.

This is achieved through a common vision, alignment, and standardization not only at the level of strategies but also regarding all activities governed by them: "Typically, strategy is the only discipline that is cultivated at the enterprise level. Strategy generates direction for the other disciplines, but then these are executed at the project level, leading to both duplication of effort and a lack of overall coherence. Now planning and implementation functions are being extended across the enterprise as well, including IT. EA provides the means to apply basic business disciplines within the IT organization. It must align enterprise activities and priorities with business strategy." [META02]

In order to facilitate the alignment of the different artifacts the EA is made up of, a common understanding of the EA is necessary to provide a communication basis for the various stakeholders involved in the management process. Accordingly, Lankhorst [La05] provides a definition of EA with a focal point on priciples, methods, and models: "Enterprise Architecture (EA) is a coherent whole of principles, methods, and models that are used in the design and realisation of an enterprises organizational structure, business processes, information systems, and infrastructure." [La05]. In addition to the static aspects as mentioned by Lankhorst, we focus on the dynamic aspects of EA management, as described in the definition stated above, to emphasize the planning aspects of the process aligning business and IT.

This alignment of business and IT concerning the IT architecture is also discussed and analyzed by Ross [Ro03], who developed a model consisting of the four architectural stages *application silo*, *standardized technology*, *rationalized data* and *modular* during which the alignment of IT and business grows: "The payback for enterprise IT architecture efforts is strategic alignment between IT and the business. Alignment will generate a higher return on the firm's IT investments and focus the firm's project portfolio on initiatives likely to have strategic impact." [Ro03]

3.1.1. Starting an EA Management Endeavor – The *EAM Pattern Catalog*

As alluded to above, EA management is one of the major challenges of modern enterprises. Whatever preliminary work exists in a company, there is always a demand for a more structured way to start an EA management endeavor or to carry on the existing EA management process. However, the variety of approaches existing in academia and practice exhibit at least one of the following problems:

- EA management is introduced from scratch, not considering related initiatives already present inside or outside the organization.
- EA management frameworks, like Zachman [Za92], TOGAF [TOG08], etc., are usually either too abstract and therefore not "implementable", or too extensive to be used in real world, as they have to be utilized as a whole.

- Lacking an actual starting point for the EA management initiatives, companies tend to call for proposal to a wide number of potential EA stakeholders. Consolidating their demands and integrating their information needs, an all-embracing EA management approach is likely to develop, which would demand a vast amount of data to be gathered, although only a part of it would be needed to address the actual pain points of the company.
- If an approach has been implemented, it is mostly not documented, why certain decision have been taken, e. g. why a special entity has been introduced to the information model. This leads to information models, which cannot be adapted or extended due to the fact that no one knows what aspects rely on which parts of the model.
- Approaches proposed e. g. by organizations or standardization groups are usually a *complete or nothing* approach, meaning that it is supposed to be introduced as one single piece instead of an incremental introduction.

This results in EA management approaches that cannot evolve according to the maturity level of the company. In order to address the problems stated above, we resort to the idea of patterns, well known from other disciplines like architecture or software engineering. EAM patterns [Bu07b] describe possible solutions for recurring problems that can and may have to be adapted to a specific enterprise context.

The *EAM Pattern Catalog* [Bu08] is a collection of best practice EAM patterns for addressing specific concerns in EA management related areas, e. g. architectural standardization, application landscape planning, or interface, business object, and service management. Thus, the *EAM Pattern Catalog* provides a holistic and generic view on the problem of EA management by providing detail and guidance needed to systematically establish EA management in a step-wise fashion within a given enterprise. The *EAM Pattern Catalog* contains the following three kinds of EAM patterns:

- A Methodology Pattern (M-Pattern) defines steps to be taken in order to address given concerns. Furthermore, as a guidance for applying the method, statements about the intended usage context are provided, which include the concerns to which the M-Pattern can be applied. These concerns are addressed by procedures defined by the M-Pattern, which can be very different, ranging from e.g. visualizations and group discussions to more formal techniques as e.g. metrics calculations.
- A Viewpoint Pattern (V-Pattern) provides languages used by M-Patterns. A V-Pattern proposes a way to present data stored according to one or more I-Patterns. In our research project *software cartography* (see e. g. [LMW05, Bu07a, Bu07b, Wi07b]), we found that industrial users often specify viewpoints by example. This means that an exemplary view is provided for the viewpoint, possibly together with some textual explanations. While we do not contend that this may be sufficient in certain use cases, e.g. sketching concepts in presentations, we see problems, when the goal is providing official information to a wider audience for an extended period. In order to ensure the understandability of a view, we regard a legend to be mandatory.
- An Information Model Pattern (I-Pattern) supplies underlying models for the data visualized in one or more V-Patterns. An I-Pattern contains an information model fragment including the definitions and descriptions of the used information objects.

The *EAM Pattern Catalog* can be utilized to introduce a problem-adequate, organization-specific endeavor to EA management based on best practices. Thereby, the *EAM Pattern Catalog* supports both a green field approach and the enhancement of an already established EA management process. In the case of a green field approach, first of all the pain points of the company, the so called concerns have to be identified. This is supported by the list of concerns included in the *EAM Pattern Catalog*. The selected concerns include references to M-Patterns that can be used to address these concerns. The methodology described in the M-Pattern uses certain V-Patterns for visualizing aspects of the EA, which are referenced by the M-Patterns. Based on the selected V-Patterns the needed I-Patterns have to be selected. The last step is to integrate the EAM patterns to a organization-specific approach for EA management. This green field approach to EA managementis illustrated in Figure 3.1.

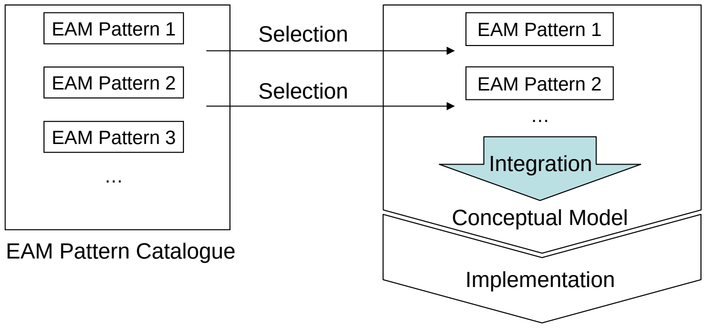


Figure 3.1.: Implementing an EA management approach based on EAM patterns

The second usage scenario for the *EAM Pattern Catalog* is to take it as a reference book for suggestions or an assessment concerning the approach currently selected in a company. This offers the possibility to compare the own EA management approach with best practices in use elsewhere. The *EAM Pattern Catalog* can e. g. be used to look for typical concerns, which occur in other companies.

3.1.2. Gathering information for EA Management

In the area of EA management it is not sufficient to collect the information ad hoc and try to use it later: Information may be outdated shortly after collecting and then an analysis would reside on outdated information. The main problem – ensuring the availability of up-to-date information – arises due to the different information suppliers needed and the high effort for a consistent set of information objects. As EA management concerns different functions from both IT and business areas, business process modelers, IT system owners, IT architects, project managers, etc. must supply the EA management process with the information needed for addressing the concerns of different stakeholders.

Figure 3.2 shows that the needed information for the EA management process is already partially present in existing information systems in organizations. Of course unification of the information into an abstracted model is necessary in order to create a coherent view.

- The information about business processes, their process flows, process steps, etc. already resides in business process modeling tools like ARIS Toolset (from IDS Scheer AG), Corporate Modeler (from Casewise Inc.), MEGA Process (from MEGA), etc.
- The information about applications, their components, interfaces, etc. resides in software modeling tools like Rational Software Architect (from IBM), Together Control Center (from Borland), etc.
- The information (infrastructure) services, their service level agreements, service user, etc. are available in a database corresponding to frameworks like ITIL [OGC00] from OGC, MOF [Mic05] from Microsoft, etc.
- The information about assets and systems, connections between hardware systems and information systems, up- and down-times, etc. is collected in systems like Open View (from HP), Tivoli (from IBM), SMS (from Microsoft), etc.
- The information about project planning and business intelligence is e. g. available in SAP Business Warehouse (from SAP), MS Project (from Microsoft), etc.

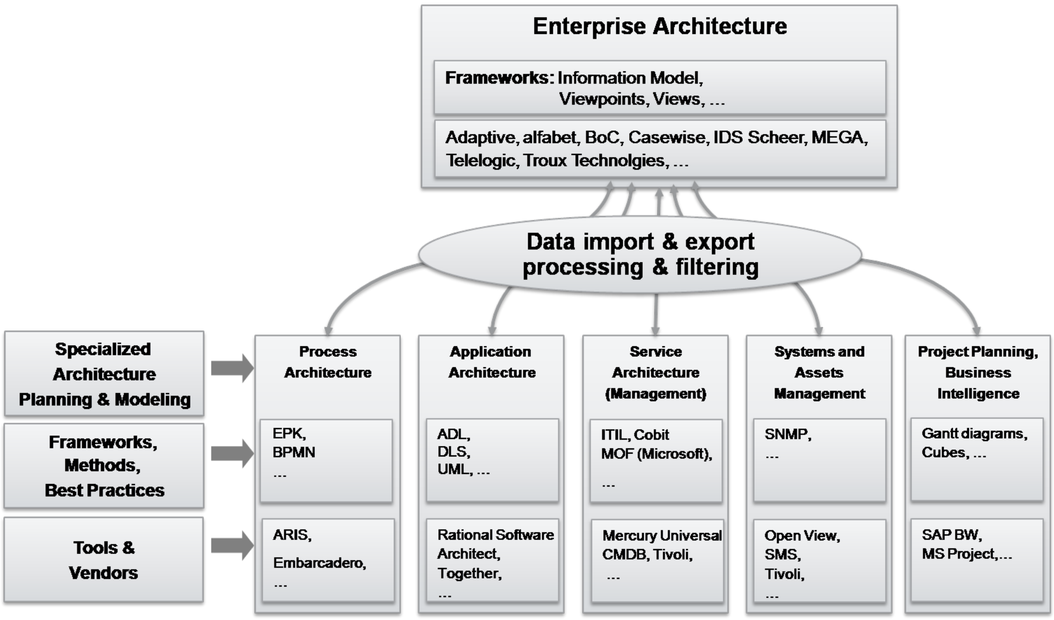


Figure 3.2.: Information filtering & processing

Whereas most of the information gathered during the EA management process is specific to the enterprise under consideration, like the process or the application architecture, there is some information, which is common to most companies, e. g. lifecycle and support times for operating systems. These data could e. g. be provided by a platform via the internet, which allows the exchange of infrastructure information between EA management practitioners and software vendors. Thereby, the information exchange is not only limited to an initial download of data but can be also leveraged to keep the information up-to-date.

As this automated information update can only be realized for infrastructure information, a continuous EA management process must ensure that the process itself is linked to related, existing management processes. Thereby, ensuring to be backed on already existing data pools. Using the data abstracted and unified from the areas described above, the EA management process develops the coherent vision and uses it to steer IT and business in an aligned way.

3.1.3. Integrating existing Management Areas

As mentioned above, the EA management endeavor must be linked to related management processes to use their input and output to plan, control, and monitor the evolution of the application landscape. Figure 3.3 shows the different management areas and illustrates their interaction.

The management areas as illustrated in Figure 3.3 are subsequently sketched and linked to the scenarios used for the evaluation of the different EA management tools (cf. Section 4.2):

- **Enterprise architecture management** constitutes the *glue* between the different management areas by providing current, planned, and target landscapes to align, plan, and control the evolution of the application landscape. The scenario *landscape management* is concerned with the creation, maintenance, and development of these views on the application landscape.
- **Demand management** forms the entry point for new action items, which may change the EA. Thereby, typical tasks of demand management are gathering demands, identify affected

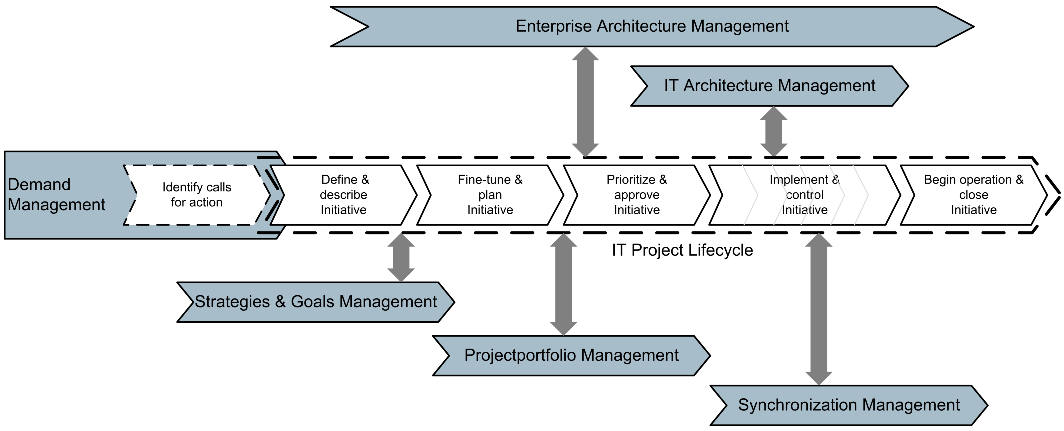


Figure 3.3.: Integrating related Management Areas [Wi07a]

architectural elements and prepare the transformation into project proposals. A scenario of the same name exists to analyze the provided tool support.

- **Strategies and goals management** is concerned with the realization, monitoring and evaluation of defined strategies and goals. Thereby, demand, projects, and their affected architectural elements are evaluated according to the strategies and goals defined for the enterprise. A scenario of the same name exists to analyze the provided tool support.
- **Project portfolio management** covers the management process concerned with retrieving the project portfolio for the next planning period according to a given budget. A scenario of the same name exists to analyze the provided tool support.
- **Synchronization management** deals with the synchronization of projects, according to their interdependencies derived from the objects (e. g. application systems or services) a given project is likely to change. Thereby, a delayed project may affect other projects, which need to be identified and rescheduled. A scenario of the same name exists to analyze the provided tool support.
- **IT architecture management** is concerned with standardizing the application landscape by defining architectural blueprints for the development of application systems. The goal of IT architecture management is to reduce the heterogeneity of the application landscape. A scenario of the same name exists to analyze the provided tool support.

The Figures 3.2 and 3.3 illustrate two different views for integrating related management tasks of EA management. Figure 3.2 focuses on the integration of existing data resources into an EA view, while Figure 3.3 shows how different management processes can be linked into a continuous and integrated process.

3.1.4. EA Management Layers & Cross Functions

Asking which concepts are of importance in EA management, practitioners as well as EA frameworks tend to enumerate a multitude of elements, which should be used in organizing the EA and be considered in establishing an EA management process. Nevertheless, no comprehensive and integrated information model connecting and reflecting all these aspects is provided. This might be the case due to a variety of causes – to name a prominent one, due to different levels of detail concerning the information regarded necessary for performing EA management. To exemplify this, there are some practitioners thinking of information on technical infrastructures, e. g. servers being valuable and important in EA management, while others abstract from this technical information towards e. g. a more machine independent view on infrastructure services.

In evaluating EA management tools, we have to rely on certain information on the company's EA to be present and usable for performing specific analyzes and answering the questions as raised in the different scenarios. Therefore, an integrated information model for a fictional department store, called *SoCaStore*, has been created. Before we detail on this actual model (cf. Section 3.2), we describe on the basic structure behind this model, giving an abstract overview of the general aspects an EA information model should take into considerations. These aspects are further refined to *layers* and *cross functions*, reflecting the elementary domains, we see constituting an EA information model.

Therein, the layers mirror the overall business-to-infrastructure makeup of the company's EA, further introducing two service layers, which can be used to describe infrastructure or application functions on an abstracted level. Complementing the layered structure containing the basic EA concepts, orthogonal cross functions are introduced. These functions are used to organize concepts, which are not part of, but can exert influence on any of the elements as organized in the layers. As an exceptional cross function, the *Measures & Metrics* are introduced, which are used to quantify aspects of EA concepts on the different layers and cross functions.

Figure 3.4 shows the different layers and cross functions. The color coding further indicates the variability concerning the involvement of information from the specific column in a holistic and compulsory EA management information model. While it is largely undisputed, that information from all the layers and cross functions is of importance for executing EA management, it might depend on the executing company, to which level of abstraction this information in detailed, as indicated by the depth of coloring.

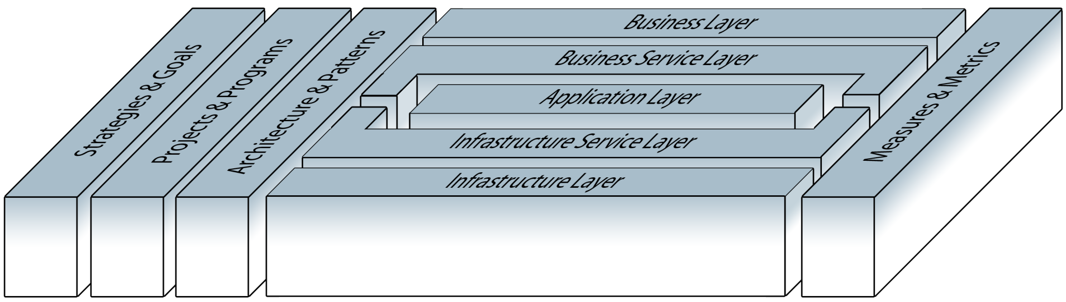


Figure 3.4.: Layers & cross functions of an information model

Having introduced the layers and cross functions, we subsequently further substantiate them, giving indications of typical concepts employed on the respective layer and providing exemplary questions, which are of importance in the layer.

The focus of the *Business Layer* lies on *Products*, *Business Processes*, *Organizational Units*, or *Lines of Businss*, etc. Exemplary questions on this layer could be:

- Which products are produced in which lines of business?
- Which business processes, e. g. for purchasing, production, or distribution, are related to the respective products?
- Which organizational units are participating in the business process execution and how are they linked?
- Which primary and supportive business processes rely on by which business functions?

The *Business Service Layer* establishes a loose coupled link between the business layer and the application layer providing business support. This link is realized using *Business Services*[2], which are

[2]These *Business Services* are commonly alluded to a *Services* especially in the context of Service oriented

further detailed by their related *Service Level Agreements (SLAs)* as well as the exchanged *Business Objects*. Exemplary questions related to this layer are:

- Which business services are used in the execution of which business processes?
- Which business objects are created, read, updated, or deleted by which services?
- Which business services exert data sovereignty for which business objects?
- What service level agreements are assigned to which services?

The *Application Layer* is concerned with the *Application Systems* realizing business support as well as their *Interfaces* and *Components*. Exemplary questions related to this layer could be:

- Which applications systems are employed in realizing which business services?
- Which interfaces and information flows exist between which application systems?
- Which infrastructure services are used by which application system?
- What are the components, an application system is made up of, where are the components hosted at?

The *Infrastructure Service Layer* is used as an intermediary layer to remove the direct coupling between the application layer and the infrastructure layer. Therefore, the concept of the *Infrastructure Service* is introduced, complemented with technical *Service Level Agreements (SLAs)*. Exemplary questions on this layer are:

- Which infrastructure services are available?
- Which application system uses which infrastructure service?
- Which dependecies between the infrastructure services exist?
- What service level agreements are assigned to which services?

The *Infrastructure Layer* describes the core *Infrastructure Systems*, e. g. middleware and hardware systems, which are used to provide the infrastructure services as outlined on the layer above. Here considerations as the following are made:

- Which infrastructure systems are employed to provide which infrastructure services?
- Which dependencies between the infrastructure systems do exist?
- Which locations are the infrastructure systems hosted at, which organizational units are in charge of maintaining?

The cross function *Strategies & Goals* is concerned with business and IT *Strategies*, which are subsequently operationalized by *Goals* and more detailed *Action Items*. Concerning this cross function, different questions could be asked, e. g.:

- Which goals can be derived from which strategy and which action items link to them?
- Which strategies result (transitively via which goals) in which demands and projects?
- Which items on which layer are affected by which goal?

The cross function *Projects & Programs* further operationalizes the goals and strategies by actual demands and projects implementing them. The concepts modeled in this cross function reflect the drivers of the actual change evolving the EA. Based on this questions as the following can be derived:

- Which action items have led to which project proposals, which projects have then been derived?
- Which project affects which items?

Architectures (SOA).

- Are the projects consistent with the goals?
- Which demands are of *run the company*, which of *change the company* type?
- What dependencies exist between the projects, which concepts caused them?

The core topic, the cross function *Architecture & Patterns* is concerned with, is the management of *Reference Models* and *Architectural Blueprints*, which are used to increase standardization in a company. Typical questions related to this cross function could be:

- Which reference models for business processes are used throughout the enterprise?
- Which architectural blueprints have been established and which solutions have been created implementing them?
- Which business processes are consistent with which reference processes?
- Which application systems rely on which architectural solution?
- Are the planned and ongoing projects consistent with the architectural standards and business reference models?

Complementing the cross functions and layers introduced above, the cross function *Measures & Metrics* contains means to quantify aspects of importance for EA management. The *Measures* and *Metrics* introduced thereby are commonly assigned to an information object from the other layers or cross functions, further providing numerical indicators for certain properties. Such quantification is common for many domains in business and IT, e. g. accounting or software development. The cross function presented here aims to extend this to objects from the EA. Thinking of exemplary questions concerning with this cross function, one could think of the following:

- Which measures do exist for which information objects?
- How are the measures quantifying a certain property?
- What are the prerequisites for a metric to be significant for a certain property?
- How does a project affect application availabilities=?
- Is the application landscape flexible?

For more details on a quantitative approach to EA management, see Section 3.1.5. The layers and cross functions as introduced above are used to organize the information about the EA likely to be collected, maintained, and analyzed in the course of a typical EA management endeavor. Nevertheless, not every company will use information from all these layers or cross functions, nor will the level of abstraction applied in each column be the same. Therefore, the information model (cf. Section 3.2) as used for the evaluation of the tools has to be regarded an exemplary one, reflecting the layers and cross functions to a level as demanded by the evaluation criteria. As these criteria have been risen by the sponsors and partners of the survey, the information model can nevertheless claim to be incorporating a rich set of information requirements from EA management practice.

3.1.5. Quantifying the EA – Measures and Metrics

Figure 3.4 introduces a cross function which can basically be of use in all of the management areas outlined in Section 3.1.3: Measures and Metrics, concerned with quantitative information about an application landscape.

This includes

- lagging indicators, metrics derived in the acutal operation of an application landscape (e.g. failure rates for specific spaces in time, average or peak loads, numbers of change requests or the effort induced by them, ...)

- leading indicators, assumed drivers of the lagging indicators. As an example, architectural and technological homogeneity might influence change efforts, or failure rates calculated on a model of the application landscape could be used as an indicator for real failure rates. This can be visualized in Figure 3.5, which relies on a model estimating availabilities and impacts of failures from information about dependencies between business applications. It colors the rectangles representing the application rectangles to represent their estimated availability, and sizes them to indicate, to what extent they affect the application landscape in kind of a failure according to the model. Hereby, a bigger rectangle represents a higher impact [LS08].

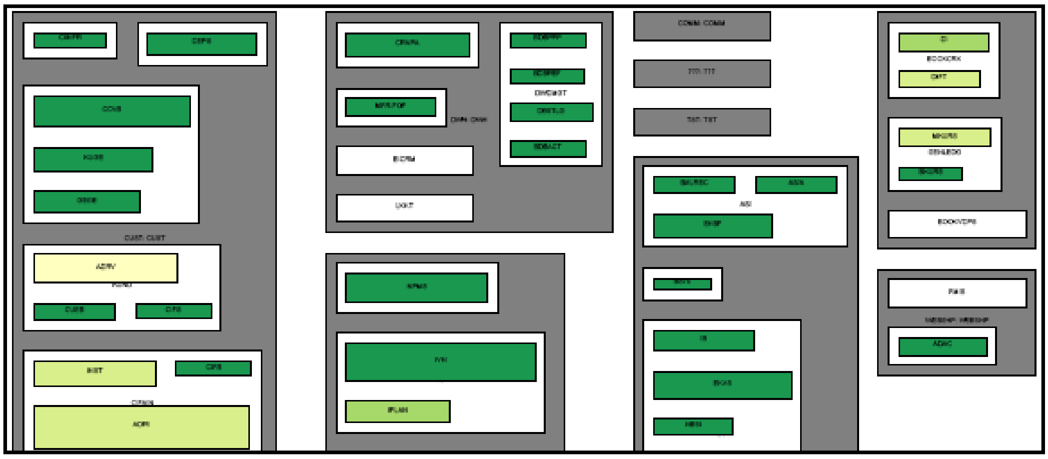


Figure 3.5.: Visualizing availability and failure impact metrics on a software map

Using quanitative techniques introduces a whole new set of approaches and possibilities into EA management, which includes:

- Providing systematic decision support based on facts processed according to an explictely defined technique, instead as solely relying on intuition of stakeholders. In the example of Figure 3.5, a detailed application landscape model, and the assumptions of the availability estimation technique are used. Thus, metrics can push decision making from religious wars and hidden agendas into the application of explicit, founded decision techniques.
- Metrics, especially if appropriately visualized, can be employed to communicate facts about the application landscape. In this context, business stakeholders value the ability to communicate the implications of specific structures or changes in an application landscape without delving too deeply into technical details. Taking again up the example of Figure 3.5, it would be hard to communicate with business about a plethora of dependencies between applications. However, the metrics visualization of the software map easily conveys *one* specific implication of the dependencies: certain business applications affect others in kind of failure, and therefore specific applications have an increased probability to fail.
- Setting goals and controlling their achievement is another use case which metrics support, similar as with the balanced scorecard. While current practices, e.g. in the field of IT controlling, might use cost information, or performance metrics as the lagging indicators introduced above, architectural metrics at application landscape level are rarely used. This might very well be a contributor to a lack of communication between business (in need of quantitative information) and IT (concerned about architecture).

As far as those approaches might seem from current practices in the field, they point in a direction where EA management is less an art practiced by IT staff, but a science providing documented, predictable approaches to experts supporting business with the IT it needs. Building the EA management pattern catalog constitutes a key advancement in this direction to us.

3.2. Information Model

In order to implement a tool support for EA management, an understanding of the EA as well as of the concepts contained therein is necessary. Data that should be modeled has to conform to a metamodel, which we call *information model*:

> An *information model* is a model, which models the relationships and attributes of information objects using the concept of information classes and associations between information classes. An information object is an abstraction of a real existing object (e. g. "Accounting:Business Process", "BMW 318i:Product", "Accounting system:Application System") aggregating the relevant information.

An information model for EA management consists of the relevant information classes, their attributes, the associations between information classes, and an additional glossary collecting specialized terms and their meanings.

Many EA frameworks (Zachman [ZIFA08], TOGAF [TOG08], DoDAF [DoD08], ...) are making suggestions for *organizing an enterprise architecture*, *establishing an EA management process*, and *connecting the artifacts into one framework*. But none of the frameworks mentioned provides an integrated information model connecting the different layers and cross functions of EA management. Therefore, we built an information model for this survey addressing the concerns of the scenarios, which

- integrates the layers and cross functions (detailed above) into one model,
- is built on a modular basis, and
- distinguishes between optional and mandatory information.

The information model for this survey reflects the typical EA management tasks as outlined in Section 4.2. In order to leverage a common understanding of the information model object-oriented modeling is used here.

As a core class the `InformationObject` is introduced holding the attribute `id`, a mandatory attribute representing a globally unique identifier of the corresponding object.

As some common attributes shared among all classes in the information model exist, these attributes have been introduced in superclasses as shown in Figure 3.6. The class `NamedObject` is thereby regarded as the common superclass for most of the classes contained in the information model, such that these classes have the following attributes:

- `name` This attribute contains the name of the corresponding object, e. g. "Headquarter". This attribute is mandatory.
- `description` This attribute holds an optional description of the corresponding object.

Figure 3.7 shows the information model as used in simulating the scenarios in this survey. All classes contained therein are subclasses of `NamedObject`, if not stated otherwise. The elements contained in the model, i. e. the classes, attributes, and associations are according to [Bu08] explained below:

`ApplicationSystem` An application system is a software system, which is part of an information system of an organization. An information system is according to Krcmar [Kr05] understood as a sociotechnical system, which is, besides the software system, made up of the infrastructure the software system is based on, and a social component, namely the employees or stakeholders concerned with it. Thereby, infrastructure and social component are not considered as belonging to the application system. Here, the application system is restricted to applications that at least support one business process[3] of the respective organization. The application system class maintains a relationship to the architectural blueprint intended for the system (see reference `intendedBlueprint`).

[3] An actual deployment of an application systems performs actual support for a business process then.

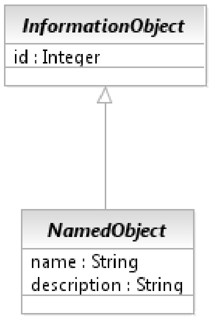


Figure 3.6.: Root classes of the information model of *SoCaStore*

`ArchitecturalBluePrint` An ArchitecturalBluePrint describes the abstract architecture of an application system using different blueprint elements (referenced by `consistsOf`), which are the building blocks of the blueprint. The architectural solutions, which conform to the blueprint are referenced by `conformsTo`.

`ArchitecturalSolution` By choosing actual solution elements for realizing the blueprint elements an architectural blueprint `consistsOf`, an architectural solution is created. This solution holds a multivalued attribute, outlining specific `properties` of the solution, such as `high availability` or `scalability`.

`BluePrintElement` A bluePrintElement describes an abstract build block of an architecture, not emphasizing on the actual implementation technology. An actual solution element, as referenced by `realizes`, gives one implementing technology for the abstract building block.

`BusinessObject` A BusinessObject represents a business entity (e.g. an invoice) that is used during the execution of a business process, which performs operations (CRUD) on the BusinessObject.

`BusinessProcess` A business process can, according to Krcmar [Kr05], be defined as a sequence of logical individual functions with connections between them. Disterer et. al. [DFH03] states input and output factors and a defined process objective as important characteristics of a process. The process class here should not be identified with single process steps or individual functions, but with high-level processes at a level similar to the one used in value chains. A business process in this context can be part of a larger, encompassing process (*super-process*) and can be have smaller, constituent processes (*sub-processes*). Additionally, a process, as seen on the level as stated above, can be part of a value chain, which is an (at least partially) ordered sequence of processes. Thus, a process can have a predecessor process and a successor process, expressed by the order relationship.

`CommunicationFunction` This class represents a special kind of function manipulating a business object. This function employs an interconnection (referenced by `makesUseOf`), via which the business object is exchanged during the business support.

`Demand` In changing the application landscape, demands for changes can be issued by different persons (referenced by `issues`). The changes stated by a demand can affect different elements of the business support (referenced by `affects`) and can be supportive for goals of the organization, indicated by the `canSupport` relationship. A demand has attributes to indicate the `expectedBenefit` and the `urgency`.

`DeployableElement` An deployable element describes an element that can be deployed at an `OrganizationalUnit`, which is responsible for the deployed element.

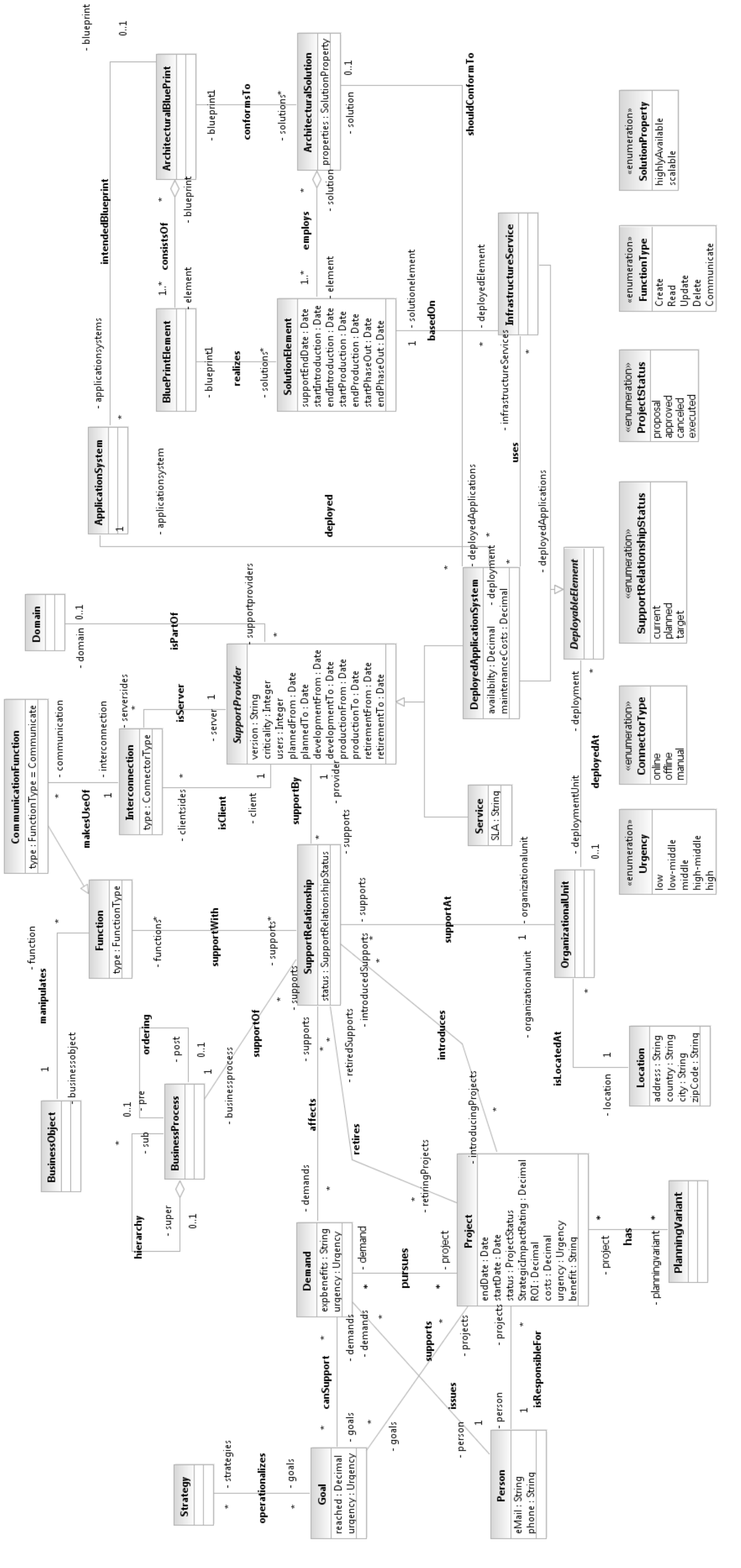


Figure 3.7.: Information model of *SoCaStore*

`DeployedApplicationSystem` In a classic (non service-oriented) architecture the support providers are realized by actually deployed application systems - therefore, deployed application system is a subclass of `SupportProvider`. Additionally, as the concept of deployment is conveniently addressed by the deployable element, the deployed application system also inherits from this type and thus maintains a `deployedAt` association to the hosting organizational unit. A deployed application system is furthermore interpreted as a deployment of an application system, as referenced by the `deployed` association. In addition, the deployed application system maintains an association to the actual infrastructure services it `uses` in performing its business support as well as to the architectural solution (referenced by `shouldConformTo`) the deployed application system is intended to conform to.

`Domain` A domain is a logical unit in the enterprise grouping together support providers, i. e. concepts supportive to a single business division, e. g. *Distribution*. [ST07]

`Function` This class represents an operation concerning a business object as employed in actually providing business support as referenced by `supportWith`. The type of business object manipulated by the function is referenced by the `manipulates` association, while the type of manipulation performed is indicated by the attribute `type`, which can take values from an enumeration - namely `create`, `read`, `update`, `delete`, or `communicate`. This very last type of function is limited to be used in a specialized class of function as explained below. Function is a subclass of `InformationObject` instead of `NamedObject`.

`Goal` A goal defines an objective that should be achieved and where the degree of achievement can be measured. Thereby, a goal realizes a corresponding strategy and can be `reached`, as an attribute indicates.

`InfrastructureService` Externally visible unit of functionality, exposed through well-defined interfaces, and meaningful to the environment (definition from [Jo05]).

`Interconnection` The interconnection represents the fact resulting from a support provider needing the support of another provider for supplying its business support. The support provider, which needs the support takes the position of the *client* as modeled by the `isClient` association; the support provider is the *server* for this interconnection as modeled by the `isServer` association (cf. [MW04]). Additionally, the interconnection holds an attribute `connectorType` for representing the type of the interconnection - limited to the values `online`, `offline`, and `manual`.

`Location` A location represents an actual geographically locatable place one or more organizational units (referenced by `isLocatedAt`) can be located at. A location holds attributes for denoting its `address`, the `country` and `city` (supported by the corresponding `zipCode`) it is located in.

`OrganizationalUnit` An organizational unit represents a subdivision of the organization according to its internal structure, e. g. the entities showing up in an organigram can be used as organizational units. An organizational unit can serve as host for `DeployableElements`.

`Person` Person in the context of the information model represents the issuer of a demand or the driver of a project.

`PlanningVariant` In order to support different scenarios for the future application landscape and business support to exist, this class is intended as a group of interrelated projects, which are intended to pursue a single future application landscape scenario.

`Project` Actual adaptations of the application landscape are performed by projects, which can arise from demands on various ways. While a number of similar demands can be combined to one project, a single demand affecting a large part of the business support can be split up into different projects. The relationship `pursues` takes these facets into account. Every project also references a person, who `isResponsibleFor` it. A project holds a large number of attributes, e. g. for determining `startDate` and `endDate` as for indicating the `projectStatus` in one of the phases `proposal`, `approved`, `canceled`, or `executed`. Additionally, many aspects of the economic impact of a project are modeled by attributes. Finally, a project can be regarded to be part of one or more `PlanningVariants`, indicating sets of projects planned to be executed together.

`Service` In a service oriented architecture, the abstract support providers, as outlined below, are modeled as services – therefore, the service class is a subclass of support provider. A service in this respect does not represent an actual application deployment providing the business support, but is to be seen as an implementation independent functional service. The service can be further described by an attribute regarding the `SLA` (service level agreement) to be provided.

`SolutionElement` A solutionElement realizes a blueprint element by providing an actually implementing technology. Therein, the technology is related to lifecycle phases `introduction, production`, and `phaseOut` - all represented by attributes providing the start- and end-dates for these phases. Additionally, an attribute holding the `supportEndDate` is given. A solution element can actually be deployed at a specific organizational unit as basis for an infrastructure service, as referenced by `basedOn`.

`Strategy` A strategy defines a long term plan, how the organization seeks to achieve its vision and mission. Thereby, a strategy does not define immediate actions or concrete resources required. A strategy is `operationalized` by associated goals.

`SupportProvider` A support provider is an abstract concept, representing an entity, which can provide support for a business process at a specific organizational unit. In the context of the information model, actually deployed application systems can be used as `SupportProvider` instances as can services. A support provider has various attributes, a subset of them specifically concerned with the support provider's life-cycle phases as `in planning, in development, in production`, and `in retirement`. Each of this phases is delimited by two attributes, e.g. `plannedFrom` and `plannedTo` when concerning the planning phase. Additionally, a support provider holds an attribute indicating its business `criticality`, a textual information about its `version`, as well as an attribute counting the number of `users`.

`SupportRelationship` This class represents the actual support of a specific business process (referenced by `supportOf`) by a specific support provider (referenced by `supportBy`) at a specific organizational unit (referenced by `supportAt`). The concept of the support relationship has been introduced as instances thereof are affected by changes of the application landscape, which could not be modeled that in detail, if none such concept was explicated. Additionally, the support relationship instances provide the information about the actual `Functions`, the support is provided with (referenced by `supportWith`).

CHAPTER 4

Scenarios of the Enterprise Architecture Management Tool Survey

Contents

4.1. Scenarios for Analyzing specific Functionality

The following sections introduce scenarios, which are used to evaluate the tool's capabilities concerning specific functionalities. Therein, functionalitites are evaluated, which are not directly connected to the fulfillment of typical EA management tasks, but can be seen contributing to any of these tasks.

4.1.1. Importing, Editing, and Validating Model Data

This scenario is the main entry point for tool supported EA management, as it might be found in a company. Data, previously gathered with a spreadsheet tool or already present in one or more databases, has to be imported into the EA management tool. Exemplarily, in the simulation of this scenario the information about the EA is assumed to be contained in various Microsoft Excel Spreadsheets. The data to be imported is organized according to the information model documented in Section 3.2. In this scenario the following concerns are addressed:

> The department store *SoCaStore* wants to introduce a tool support for EA Management. Therefore, data previously gathered in Microsoft Excel has to be imported to the repository of the tool. The information model of the data imported is considered sufficient for EA managementin *SoCaStore*, such that the tool is demanded to support this model or to be adaptable to it. In the importing process the consistency of the data in respect to the information model should be validated. Therein, especially the mandatory attributes or relationships shall be considered. Data previously not maintained in the spreadsheet should additionally be added.

In the simulation of this scenario the following questions are answered:

- Is it possible to import the data from the Excel Spreadsheets? Does the tool support the concepts from the EA information model as outlined in Section 3.2?
- To which level does the tool support *data transformations* in the importing process, e. g. splitting comma separated values maintained in one spreadsheet column?
- Does the tool support validation of consistency in the data import; is there a warning / error, if data not conforming to the information model should be imported, e. g. data omitting values for mandatory attributes?
- Can the tool deliver a report (cf. Figure 4.1) on the objects in the model, highlighting the absence of data for certain attributes or relationships?

Id	Name (english)	Kurzname (englisch) Max 8 Zeichen	Name (deutsch)	Database	Name	Size (1(small)-5(big))	Criticality Rating (1(low)-5(high))	Availability in Q1 2005 in %	Conforms to architectur	Number of Transactions in Q1	Quota of standard software	Maintenance costs in Q1 2005	Operating costs in Q1 2005
100	Online Shop	O-Shop	Online-Shop	100	MySQL Munich	2	5	99,2	no	1.450.132	20%	4.000 €	34.221 €
200	Inventory Control System	InvCon	Warenwirtschaftssystem	200	Oracle Munich	5	4	99,0	yes	68.687	40%	2.322 €	421 €
300	Monetary Transactions System (Germany)	MoTr-D	Zahlungsverkehrsystem (Deutschland)	200	Oracle Munich	3	3	98,0	no	234.545	100%	10.032 €	1.213 €
350	Monetary Transactions System (Great Britain)	MoTr-GB	Zahlungsverkehrsystem (Großbritannien)	500	DB2 London	2	3	98,1	no	21.342.134	90%	1.344 €	3.223 €
400	Product Shipment System (Germany)	ProShip	Artikelversandsystem (Deutschland)	300	Oracle	2	5	97,7	yes	234.534	30%	3.131 €	999 €
500	Accounting System	AC	Buchführungssystem (Accounting)	500	DB2 London	4	3	99,9	no	432.432	100%	4.552 €	132 €
600	Costing System	Cost	Kostenrechnungssystem (Controlling)	200	Oracle Munich	2	3	100,0	no		40%	0 €	
700	Human Resources System	HR	Personalwesen	200	Oracle Munich	2	2	99,4	yes	4.532.451	90%	31.344 €	12.331 €
800	Data Warehouse	DataWare	Data Warehouse	200	Oracle Munich	1		98,9	yes	1.243.213	100%	2.300 €	3.231 €
900	Fleet Management System	FleMan	Fuhrparkmanagement	200	Oracle Munich	1	3	99,4	yes	151.234	3%	231 €	2.321 €
1000	Business Traveling System	BusTrav	Dienstreisemanagment	300	Oracle	1	1	99,1	yes	7.345.621	0%	0 €	4.322 €
1100	Document Management System	DMS	Dokumentenmanagement	300	Oracle	2	2	99,5	no	123.465	100%	0 €	1.331 €
1200	Supplier Relationship Management System	SuppRela	Supplier Relationship Management System	300	Oracle	2	2	99,0	yes	352.563	87%	1.234 €	2.332 €
1300	MIS (Management Information System)	MIS	MIS (Management-Informationssystem)	200	Oracle Munich	2	3	99,1	no		94%		2.311 €
1400	Financial Planning System	FinPla	Financial Planning	100	MySQL Munich	1	1	99,9	no	1.235.456	100%	0 €	3.223 €
1500	Campaign Management System	CampMan	Kampagnenmanagementsystem	300	Oracle	2	2	99,8	no	3.245.247	100%	324 €	32.434 €
1600	POS System (Munich)	POS-Muc	Kassen-System (POS-Software)	200	Oracle Munich	1		99,4	yes	3.452.346	50%	0 €	3.453 €
1620	POS System (Hamburg)	POS-HH	Kassen-System (POS-Software)	300	Oracle	1	5	98,4	yes	2.532	50%	0 €	8.765 €
1650	POS System (London)	POS-Lon	Kassen-System (POS-Software) (Großbritannien)	500	DB2 London	1	5	97,5	no	432.645	60%	4.322 €	4.548 €
1700	Price Tag Printing System (Munich)	PTS-Muc	Preiskartendruck (Deutschland/München)	200	Oracle Munich	1	3	97,3	no	3.452.365	100%	324 €	6.564 €
1720	Price Tag Printing System (Hamburg)	PTS-HH	Preiskartendruck (Deutschland/Hamburg)	300	Oracle	1	3	99,3	no	243.626		2.342 €	6.463 €
1750	Price Tag Printing System (London)	PTS-Lon	Preiskartendruck (Großbritannien)	500	DB2 London	1	3	99,1	no	4.573.657	80%	23.432 €	9.855 €
1800	Worktime Management (Munich)	Work-Muc	Zeiterfassung (für Mitarbeiter)	200	Oracle Munich	3	3	99,5	yes	1.345.234	100%	533 €	564 €
1820	Worktime Management (Hamburg)	Work-HH	Zeiterfassung (für Mitarbeiter)	300	Oracle	2	3	98,3	yes	45.656	100%	23.422 €	4.654 €
1850	Worktime Management (London)	Work-Lon	Zeiterfassung (für Mitarbeiter) (Großbritannien)	500	DB2 London	2	3	98,2	no	256.245	100%	23.423 €	6.454 €
1900	Customer Complaint System	CusComp	Reklamations-Mngmt-System	600	PostgreSQL	1	2	99,1	no	4.324.262	0%	0 €	454 €
2000	Customer Satisfaction Analysis System	CusSati	Datensammlung-/ und Analyse Kundenzufriedenheit	600	PostgreSQL	1	2	97,1	no	2.434.534		0 €	8.643 €
2100	Customer Relationship Management System	CRM	Kundenmanagementsystem (CRM)	300	Oracle	3	4	99,9	yes	4.325.423	4%	0 €	4.345 €

Figure 4.1.: Scenario *Importing, Editing, and Validating Model Data*: Report showing missing data

- Does the tool enable the user to edit data in a convenient way, e. g. by providing pick-lists?
- Does the tool provide more sophisticated editing mechanisms, e. g. by supplying an assistant for determining a priorizing attribute by performing multiple comparisons?
- Is it possible to link external documentation, e. g. present in Microsoft Word documents, to specific information objects in the tool?

4.1.2. Creating Visualizations of the Application Landscape

This scenario evaluates, if the tool is capable of preparing data about the application landscape to visualizations thereof. Therein, automatical creation of different types of visualizations is mainly considered. The concerns of this scenario are:

> The department store *SoCaStore* wants to get an overview of its application landscape and its EA. This should be accomplished by the creation of six different visualizations for different aspects of the application landscape - the first four of them covering the software maps as detailed in Chapter C: a cluster map, a process support map, a time interval map, and a graphlayout map as well as a swim lane diagram and a portfolio matrix.

During the simulation of the scenario the following questions should be answered:

- Does the tool support the creation of software maps, such as a cluster map (cf. Figure 4.2), a process support map (cf. Figure 4.3), a time interval map (cf. Figure 4.4), and a graphlayout map (cf. Figure 4.5)?
- Does the tool support visualizations as swim lane diagrams (cf. Figure 4.6), and a portfolio matrix (cf. Figure 4.7)?
- How strong is the automation in the creation of these visualizations? Are they automatically *generated* or do they have to be drawn?
- What degree of flexibility regarding these visualizations does the tool provide? Does the tool e. g. support the representation of services instead of application systems, or of domains instead of organizational units?

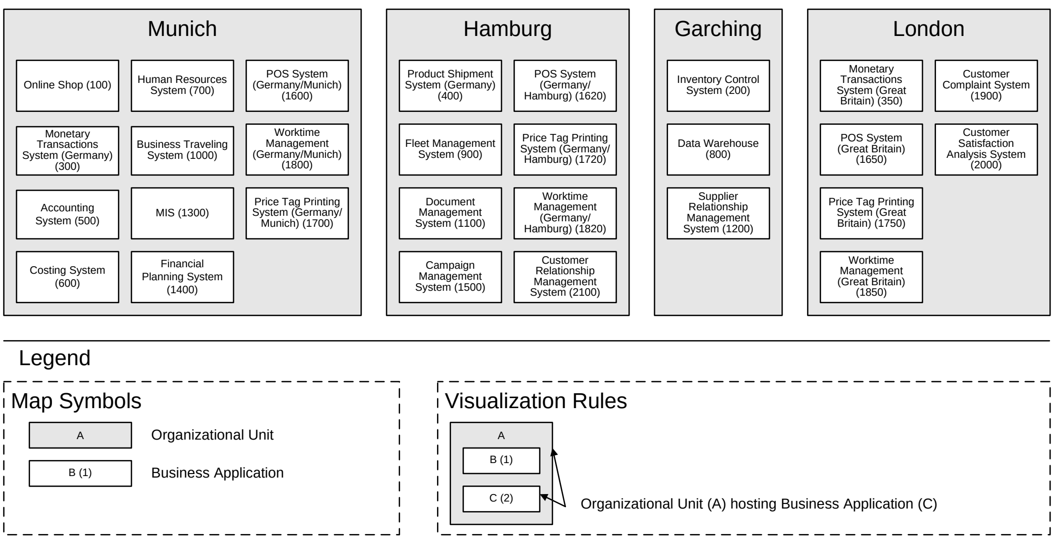


Figure 4.2.: Scenario *Creating Visualizations of the Application Landscape*: Cluster map

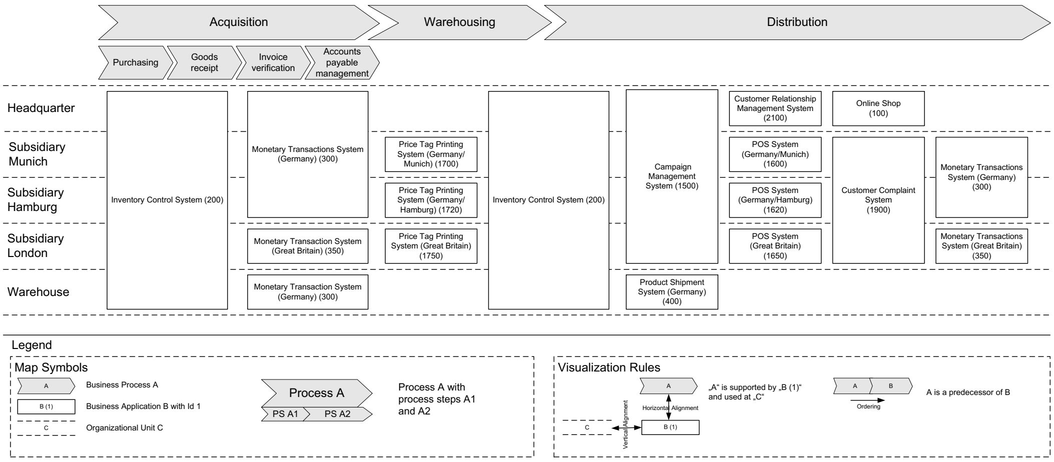


Figure 4.3.: Scenario *Creating Visualizations of the Application Landscape*: Process support map

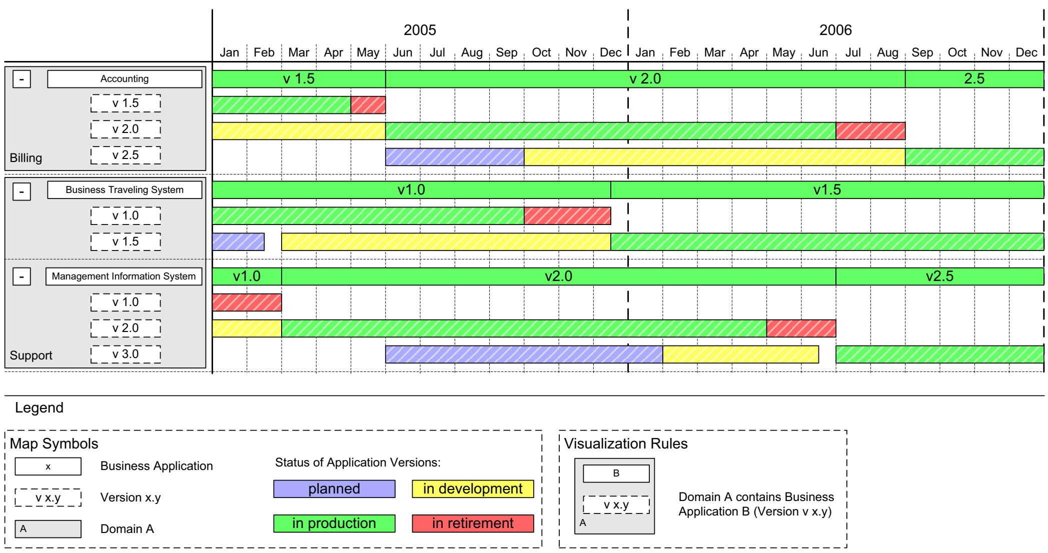


Figure 4.4.: Scenario *Creating Visualizations of the Application Landscape*: Time interval map

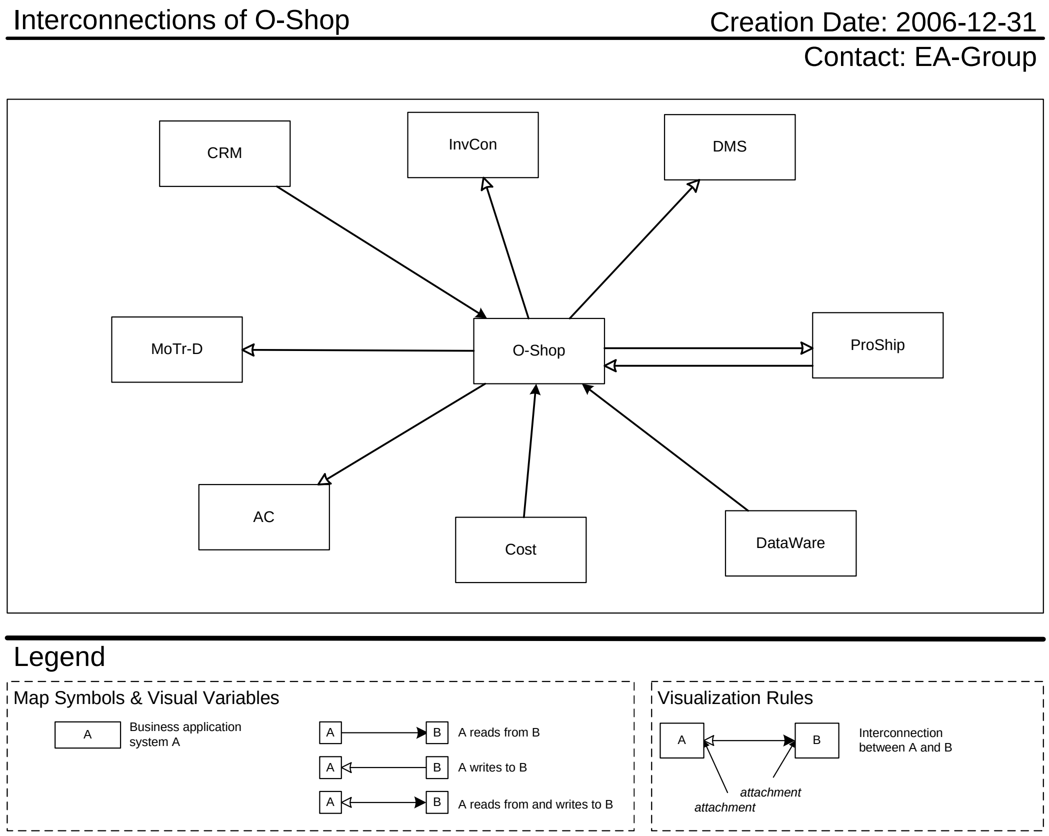


Figure 4.5.: Scenario *Creating Visualizations of the Application Landscape*: Graphlayout map

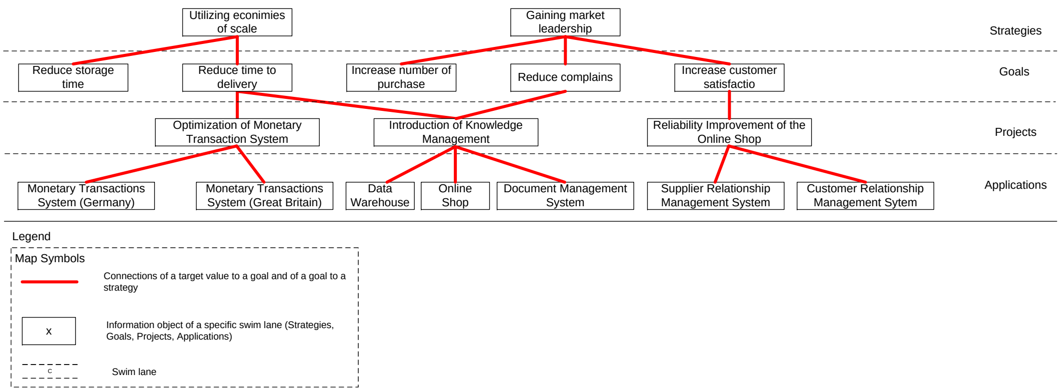


Figure 4.6.: Scenario *Creating Visualizations of the Application Landscape*: Swim lane diagram

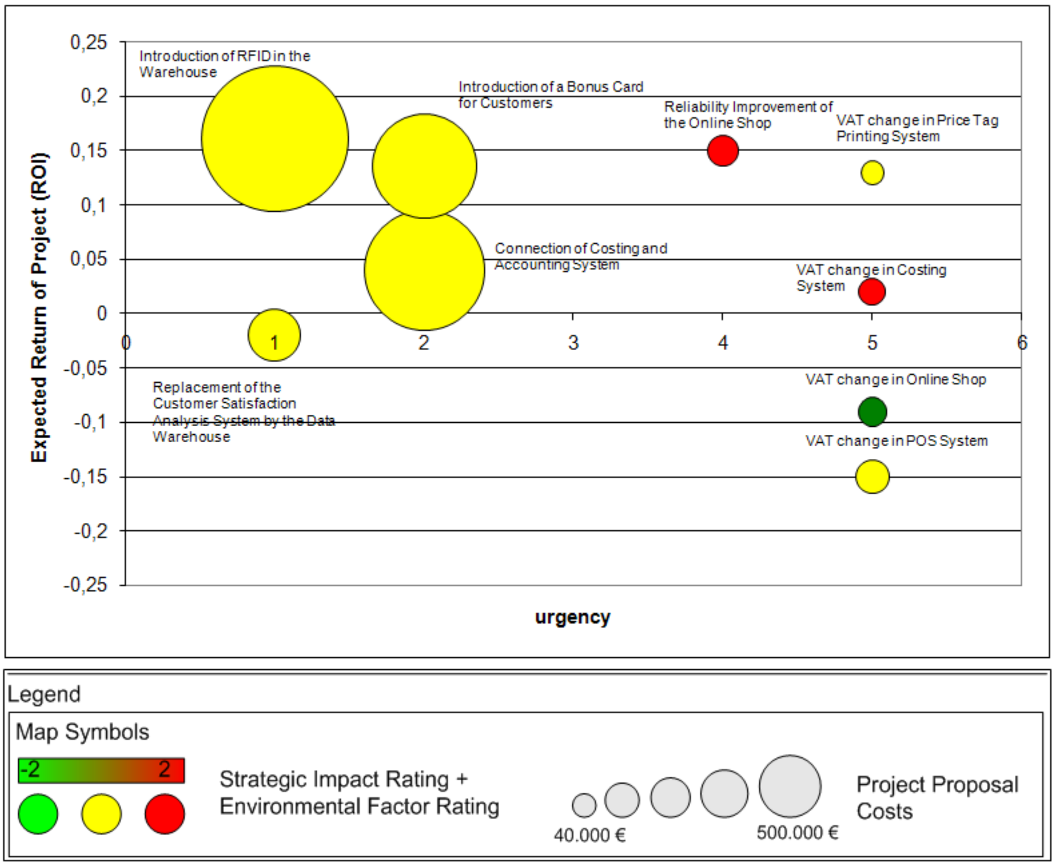


Figure 4.7.: Scenario *Creating Visualizations of the Application Landscape*: Portfolio matrix

4.1.3. Interacting with and Editing of Visualizations of the Application Landscape

This scenario evaluates the tool's support for manual adaptations of the visualizations previously generated, e. g. for resizing. Concerning this support, it is evaluated, if the tool is capable of maintaining these user updates during an automated (re-)generation of the visualization. Furthermore, this scenario deals with the tool's support for filtering and highlighting of objects in a visualization. Additionally, adaptations concerning the reports created by the tool are considered.

The concerns of this scenario are:

> In using the visualizations of the application landscape the users from *SoCaStore* want to perform adaptations on the visualizations, such as moving or resizing symbols. Additionally, for presentation reasons specific objects in the visualization should be filtered or highlighted. Furthermore, the team of *SoCaStore* wants to execute minor changes in the information about the application landscape via adaptations to the visualizations, using the tool for graphical modeling of the application landscape. Additionally, reports previously generated should be adapted, e. g. by applying *filtering*, *sorting*, or *grouping*.

In the simulation of the scenario the following questions should be answered:

- Does the tool support manual adaptations of visualization? Does the tool distinguish between adaptations, which do not, and changes, which do alter the visualization's semantics?

- Does the tool consider manual adaptations in the (re-)generation of a visualization or are the user made changes discarded?
- Does the tool provide the capability to highlight objects of interest?
- Does the tool support a filtering mechanism usable for hiding objects, which are not considered important by the user - specified, e. g. by providing filtering criteria?
- Does the tool provide the capability to *close* symbols in order to hide a complex inner structure?
- Does the tool support impact analysis, e. g. traversing from a selected element to connected elements and highlighting these elements?
- Does the tool provide navigation mechanisms in the visual model, e. g. a possibility to *jump* to a closely related object?
- Does the tool support filtering mechanisms on reports for hiding objects or attributes not considered important by the user?
- Does the tool provide the capability to perform sorting or grouping according to one or more attributes in the reports?

4.1.4. Annotating Visualizations with Certain Aspects

This scenario deals with annotating basic visualizations with additional aspects, like operating costs of application systems, in order to furthermore characterize elements of the application landscape. Preferably, this should be accomplished by extending one of the previously generated visualizations. This can be achieved, e. g. by color-coding applications in respect to their operating costs, or adding a symbol such as the traffic lights symbol in Figure 4.8 to the visualizations of systems to represent their

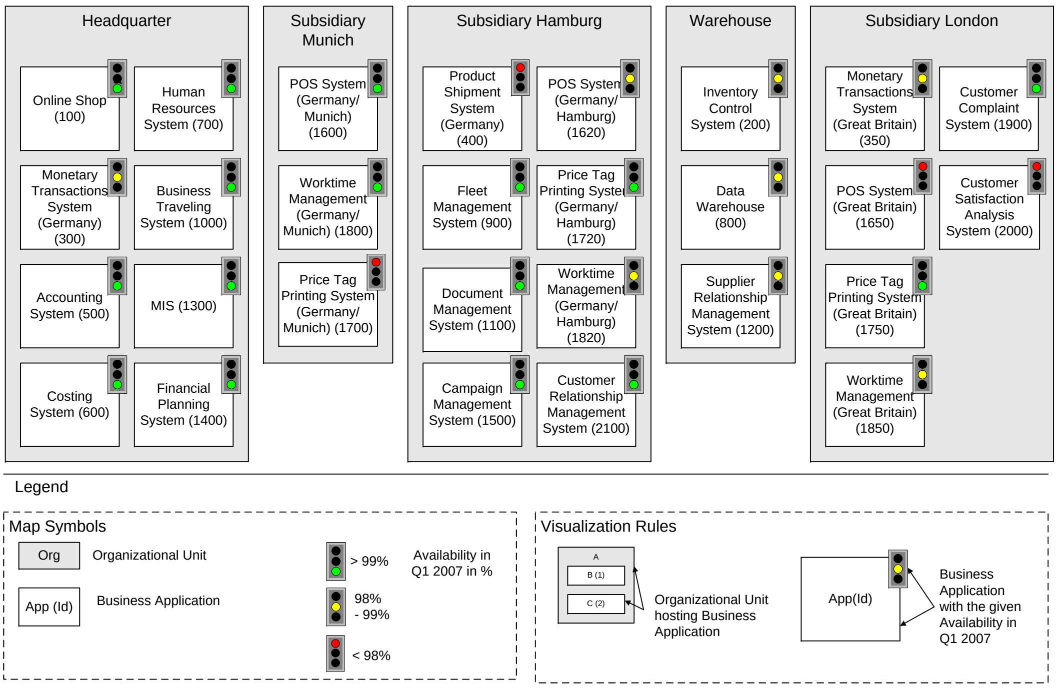


Figure 4.8.: Scenario *Annotating Visualizations with Certain Aspects*: Cluster map with availability of the application systems

availability. Additionally, compartments[1] could be used for visualizing certain aspects' values. Certain aspects, in respect to this scenario, may not only be plain attribute values but also indicators (e. g. metrics) calculated from attributes and relationships. The concerns of this scenario are described as follows:

> The department store *SoCaStore* wants to get an overview of the application systems that raise high maintenance costs. This information should be visualized additionally on an already existing visualization of the application landscape. As an additional property of the application systems, availability should be visualized on a similar map. Finally, the process support of *SoCaStore* should be analyzed regarding the usage of standard applications.

During the simulation of the scenario the following questions should be answered:

- Does the tool support the visualization of the maintenance costs of the application systems of *SoCaStore*, e. g. by changing the background color of the symbolic representations of the application systems (see Figure 4.9)?

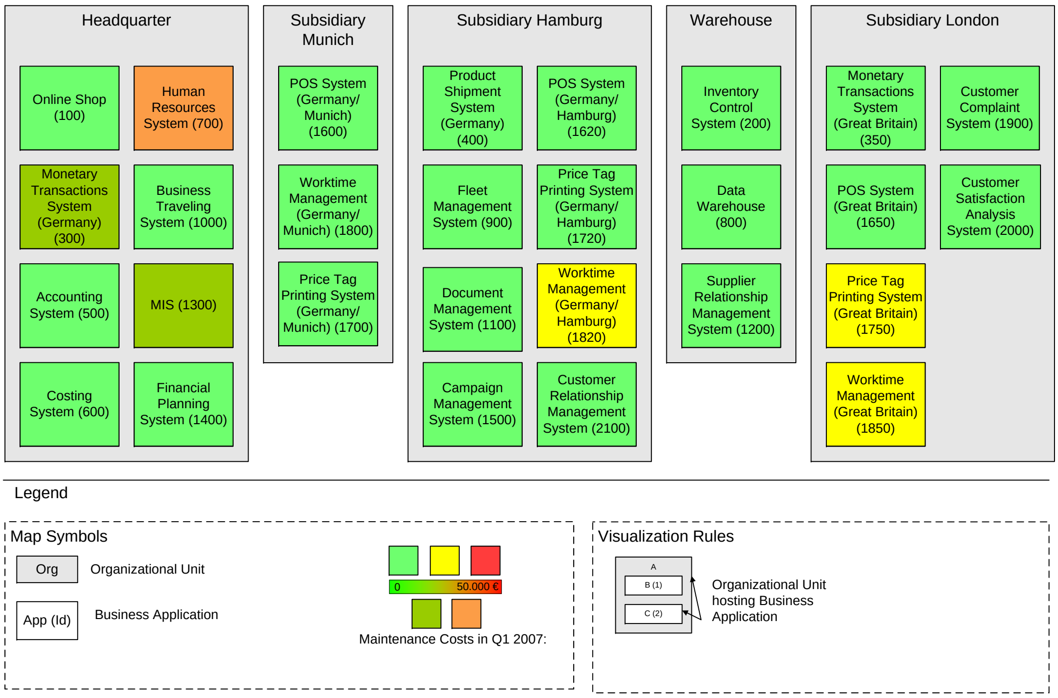


Figure 4.9.: Scenario *Annotating Visualizations with Certain Aspects*: Cluster map with maintenance costs

- Does the tool provide the capability to visualize the availability of each application system using a threshold, e. g. by adding a red/amber/green traffic light to the symbolic representation of the system (see Figure 4.8)?
- Is it possible to have the above attributes visualized together in a single visualization by combining the symbolic notations for both (cf. Figure 4.10)?
- Does the tool provide the capability to visualize the usage of standard applications, e. g. by indicating standard application systems in a visualization with a thick border (cf. Figure 4.11)?

[1] A *compartment* is a separate section in a symbol, containing additional information. The concept department is extensively used with UML class diagrams, such as the one provided in Section 3.2.

- Is it possible to create a tabular report similar to the one in Figure 4.12 showing specific aspects of each application system?

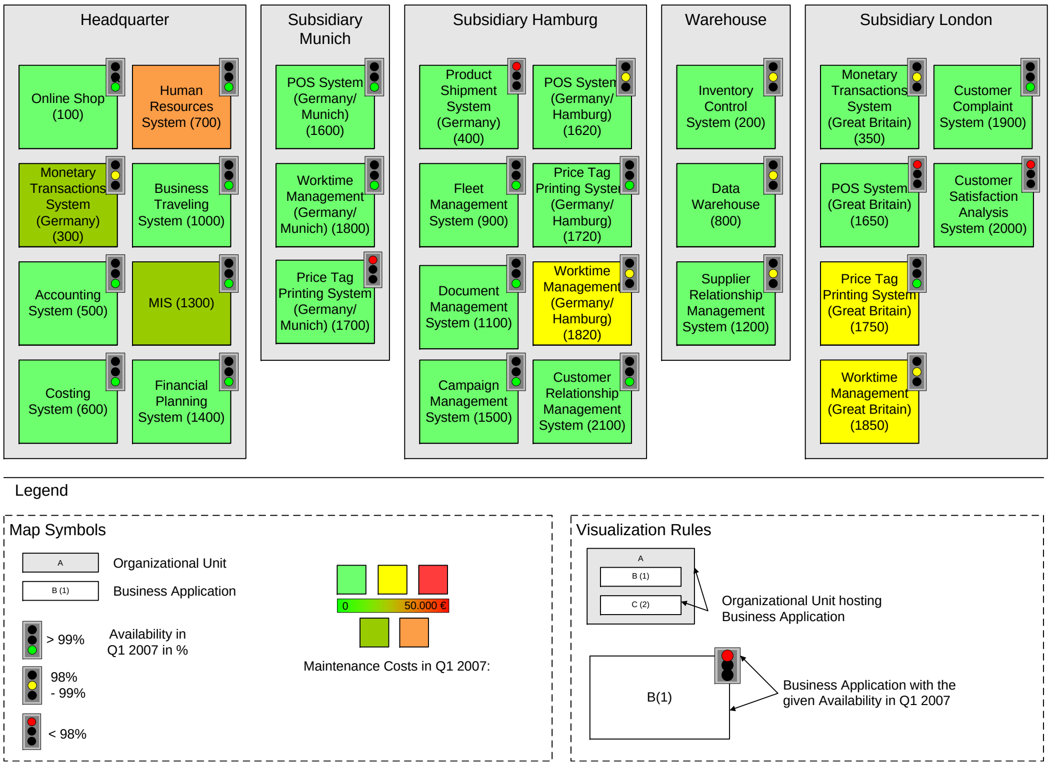


Figure 4.10.: Scenario *Annotating Visualizations with Certain Aspects*: Cluster map with maintenance costs and availability

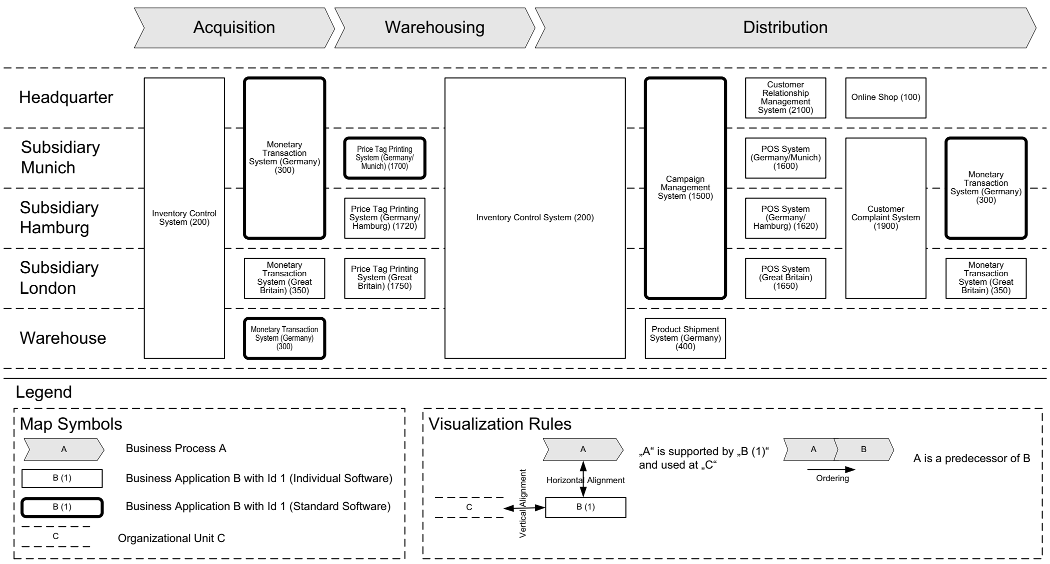


Figure 4.11.: Scenario *Annotating Visualizations with Certain Aspects*: Process support map with indication of standard applications

id	name	availability	maintenance costs	nuber of users
100	Online Shop	99,2%	4.000,00 €	50
200	Inventory Control System	99,0%	2.322,00 €	30
300	Monetary Transactions System (Germany)	98,0%	10.032,00 €	10
350	Monetary Transactions System (Great Britain)	98,1%	1.344,00 €	10
400	Product Shipment System (Germany)	97,7%	3.131,00 €	15
500	Accounting System	99,9%	4.552,00 €	10
600	Costing System	100,0%	- €	16
700	Human Resources System	99,4%	31.344,00 €	5
800	Data Warehouse	98,9%	2.300,00 €	18
900	Fleet Management System	99,4%	231,00 €	13
1000	Business Traveling System	99,1%	- €	11
1100	Document Management System	99,5%	- €	23
1200	Supplier Relationship Management System	99,0%	1.234,00 €	9
1300	MIS (Management Information System)	99,1%	13.441,00 €	14
1400	Financial Planning System	99,9%	- €	10
1500	Campaign Management (Marketing Automation System)	99,8%	324,00 €	7
1600	POS System (Germany/Munich)	99,4%	- €	5
1620	POS System (Germany/Hamburg)	98,4%	- €	4
1650	POS System (Great Britain)	97,5%	4.322,00 €	6
1700	Price Tag Printing System (Germany/Munich)	97,3%	324,00 €	7
1720	Price Tag Printing System (Germany/Hamburg)	99,3%	2.342,00 €	6
1750	Price Tag Printing System (Great Britain)	99,1%	23.432,00 €	8
1800	Worktime Management System (Germany/Munich)	99,5%	533,00 €	13
1820	Worktime Management System (Germany/Hamburg)	98,3%	23.422,00 €	12
1850	Worktime Management System (Great Britain)	98,2%	23.432,00 €	15
1900	Customer Complaint System	99,1%	- €	15
2000	Customer Satisfaction Analysis System	97,1%	- €	9
2100	Customer Relationship Management System	99,9%	- €	14

Figure 4.12.: Scenario *Annotating Visualizations with Certain Aspects*: Report showing measures of the application systems

4.1.5. Supporting lightweight Access

This scenario is concerned with methods for providing lightweight access to the tool's data. Therein, web access is considered to be the preferable access channel. In this scenario the following concerns are addressed:

> In order to promote the enterprise architecture effort and to increase the awareness for EAM among the employees, the information and visualizations gathered in the tool should be made accessible via a lightweight client, preferably a web client. Beside facilities for publishing existing visualizations, this calls for the possibilities to make these visualizations easy to use, hiding most of the features available to the modeler. The visualizations and informations in the web client should preferably be based on *live* data or should be created by a scheduled batch export. Furthermore, visualizations should be assigned *unique identifiers* by the tool, providing a way for creating a *permalink*[2] to the specific visualization.
>
> Additionally, the department store wants to distribute the effort of keeping data in the repository up-to-date to a broader group in the enterprise. Therefore, a web client enabling the user to change a subset of data, e. g. data about certain application systems, should be provided by the tool.

The tool's capabilities concerning the two aspects of lightweight access (*read-only* and *editing*) are evaluated in the scenario simulation. Therefore, especially concerning the read-only access, the following questions are answered:

[2]The concept *permalink* originates from the field of blogging, where it represents a URL, which is associated unchangingly with a specific version of a specific element. Changes to this element are not reflected in the version available at the URL, but are published under a different URL.

- Does the tool provide web access to information (tabular reports) and visualizations previously modeled?
- Is it possible to limit the access of a user to certain visualizations and information?
- Is it possible to show/hide layers in the visualizations or to zoom?
- Does the tool support navigation in the visualizations, e. g. to get from an overview diagram to diagrams with more details or tabular listings of information?
- Is it possible to search for objects in the tabular reports or visualizations?
- Is it possible to apply filtering to the tabular reports or visualizations to hide certain aspects or objects?

Concerning the editing access to the tool's data additionally the following questions are answered in the scenario simulation:

- Is it possible to edit certain object's data, e. g. of certain application system?
- Does the tool provide mechanisms to limit the editing access of a user exclusively to a certain set of objects?

4.1.6. Editing Model Data using an external Editor

This scenario reflects the need of having the possibility to use office software, like Microsoft Excel (which is used as an exemplary external editor within the evaluation), for editing the EA management data instead of using the tool's built in manipulation capabilities. This is especially of interest, if the EA management tool does not provide strong support for *batch* editing in tables instead of editing in dialogs. The concerns of the scenario are described as follows:

> Besides utilizing the tool's built-in facilities for manipulating the data, a user involved with EA management in *SoCaStore* wants to perform changes on the data about the application landscape offline - preferably using office software, as Microsoft Excel. After having performed changes on the offline data, the information should be synchronized with the data governed by the tool and a conflict resolution should be supported.

During the simulation of the scenario, the following questions should be answered:

- Does the tool support exporting data for editing to a format accessible for an office tool?
- Is it possible to lock elements from the tool's model to prevent editing data previously exported or does the tool support a *check-out* mechanism for model data?
- Does the tool provide mechanisms for re-importing the data from the export format?
- Does the tool support the user in detecting and resolving conflict states concerning data edited in both the EA management tool itself as well as in the exported data? Can the user decide on the merge strategy in combining the data?

4.1.7. Adapting the Information Model

This scenario reflects the requirement that the information model of the EA management tool should stay adaptable, even if data conforming to the model has already been imported into the tool. The concerns of this scenario are as follows:

> The *SoCaStore* department store introduces new concepts to the information model for EA management. These concepts should be supported by the tool, such that adaptations of the information model contained should be performed.

In simulating this scenario the following questions should be answered:

- Is it possible to introduce new classes to the information model? Can predefined classes be adapted, hidden, or deleted? How does the tool react to the deletion of a class, for which objects are contained in the data?
- Is it possible to introduce additional attributes? Can predefined attributes be adapted, hidden, or deleted? Are mandatory attributes as well as default values for attributes supported? Are typed attributes supported? How does the tool react to the deletion or adaptation of an attribute having already assigned values?
- Is it possible to introduce new relationships? Can predefined ones be adapted, hidden, or deleted? Is it possible to add attributes to a relationship as well as to supply cardinality constraints for relationship ends? How does the tool react to the deletion or adaptation of a relationship having already been used?
- Is it possible to create a visualization (similar to the one shown in Figure 3.7) or report of the current information model with its classes, attributes, and relationships?
- Is it possible to export the current information model and to (re-)import (previously exported) information models into the tool?
- Is it possible to assign a symbolic representation type to a class from the information model, e. g. to define that a *business process* should be represented as a *chevron*?

4.1.8. Handling large scale Application Landscapes

This scenario analyzes the tool's capabilities to handle large scale application landscapes containing thousands of application systems and interconnections between them. Therein, aspects of performance as well as of convenient tool use are considered, e. g. a smart support for picking an element from a list of several hundred entries. The concerns of this scenario are:

> *SoCaStore* wants to ensure that the EA management tool can handle the application landscape, even if an extensive growth in size and modeling granularity takes place. Therefore, exemplary data about application systems should be imported into the tool and visualized in exemplary visualizations.

The specific questions to be answered in the simulation of this scenario are:

- Can the tool import data describing an application landscape consisting of 1000 application systems[3], where each of these systems has 10 to 15 interconnections to other application systems, and bears a number of 70 different attributes?
- Can the tool create a cluster map as shown in Figure 4.13 visualizing the application landscape including all application systems and the organizational units, where the application systems are hosted, as well as the interconnections between the application systems?
- Can the tool create an aggregated visualization, as visualized in Figure 4.14, showing the aggregated interconnections between the domains of the application landscape?
- Does the tool support editing data in the large scale application landscape in a convenient way, e. g. by providing smart support for finding elements?
- Can the tool create a report containing all applications and their attributes?

[3] If the tool is found capable of handling 1000 application systems without lagging, the number is increased to 5000 or finally 10000 application systems.

In the context of this scenario, also the EA management tool's response times in handling (not in importing) should be evaluated. Thereby, only perceived duration times as *live, coffee break*, and *overnight* are stated, to give a first impression of the tool's capabilities to handle large scale application landscapes

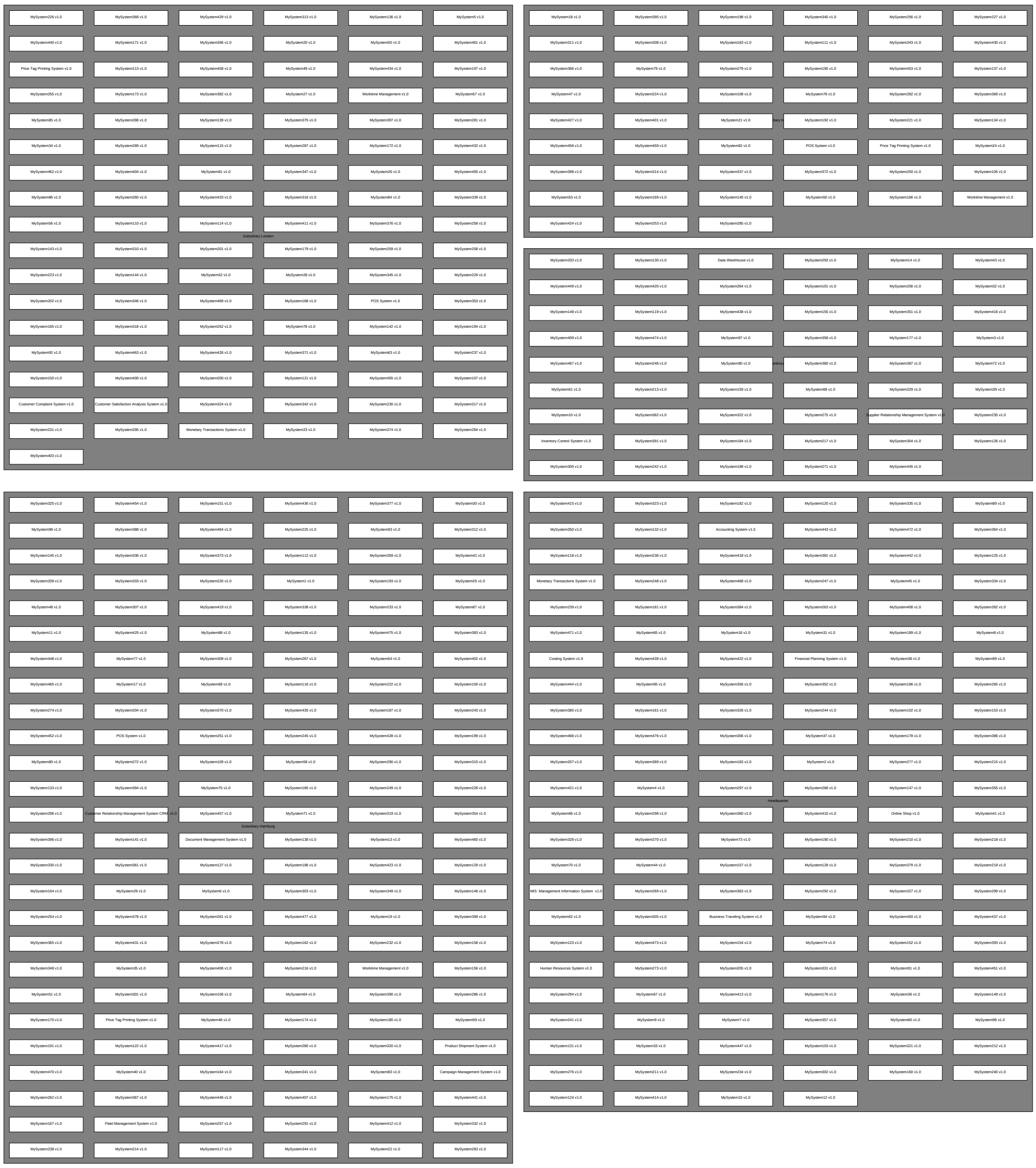


Figure 4.13.: Scenario *Handling large scale Application Landscapes*: Cluster map

4.1.9. Supporting multiple Users and collaborative Work

This scenario is concerned with methods for providing multiple users and collaboration support. Concerning this support it is evaluated if aspects like e. g. role or user based access controls on the information collected within the tool are supported. Thereby, the different levels of granularity to configure the access control such as on model-level, entity-level, and attribute-level are of interest as

Figure 4.14.: Scenario *Handling large scale Application Landscapes*: Cluster map with aggregated interconnections

well as the different kinds of access right that can be granted. Furthermore, the scenario evaluates the tool's capabilities to support notifications services to identify aged data or changes in the repository as well as workflow management. In this scenario the following concerns are addressed:

> By establishing a user management within the EA management tool the department store *SoCaStore* wants to ensure that the access to information and visualizations can be enabled according to the position of a certain user and that modifications on data or visualizations can be tracked to save audit trails. Besides, the members of *SoCaStore* department store, as e. g. the application owners would like to get notifications if certain information or visualizations are modified. In addition the *SoCaStore* department store wants to ensure that certain process steps within a workflow, as e. g. an approval, are executed by predefined users before the next step can be accomplished.

During the simulation of the scenario the following questions should be answered:

- Is it possible to allow role or user based access to information? What granularity levels are supported by the tool?
- Does the tool support role or user based windows and dialogues?

- Is it possible to limit the access of a user to certain visualizations and information?
- Is it possible to limit the access of a user to read-only mode for certain visualizations and information?
- Is it possible to save metadata (as e. g. last modification date, last modifying user) for visualizations and information to allow auditing?
- Does the tool provide support for workflow management, like documented demands must be reviewed by an IT strategist before transformed into a project
- Is it possible to setup a notification mechanism to identify aged data, which hasn't been changed for a certain time or if a predefined visualization or information is changed.

4.1.10. Scenario Simulation

The procedure for simulating the scenarios is as follows, in order to ensure the consistency of the simulation process for all tools. For every scenario in Section 4.1 the following information is gathered:

Achievement of objectives: Was it possible to create the deliverables, the scenario was aiming at (or comparable deliverables)?

Tool handling: How many pitfalls and shortcomings led to a high effort in producing the deliverables?

4.2. Scenarios for Analyzing EA Management Support

The following sections describe the scenarios used to analyze the EA management support provided by the tools. These scenarios complement the functional requirements as described in Section 4.1 with the specific EA management tasks. In the context of these scenarios the application landscape of the enterprise is modeled as a core artifact within the tool.

Thereby, different versions of the application landscape have to be considered the *current*, *planned*, and *target* version. The *current landscape* represents the status quo of the landscape as is, modeled at a certain time; *planned landscape* represents a future state of the landscape as to be at a specific time. This state is modeled by an architect at a certain time, emphasizing e. g. the changes performed by projects up to that specific future date. As a long term perspective the *target landscape* shows the architecture of the application landscape as envisioned at a certain time. There is no need to have projects defined transforming the current or planned landscape into the target landscape. Furthermore, the target landscape does not specify deployed application systems but refers to envisioned application systems. For an explanation of the relationships between the landscape versions see Figure 4.15.

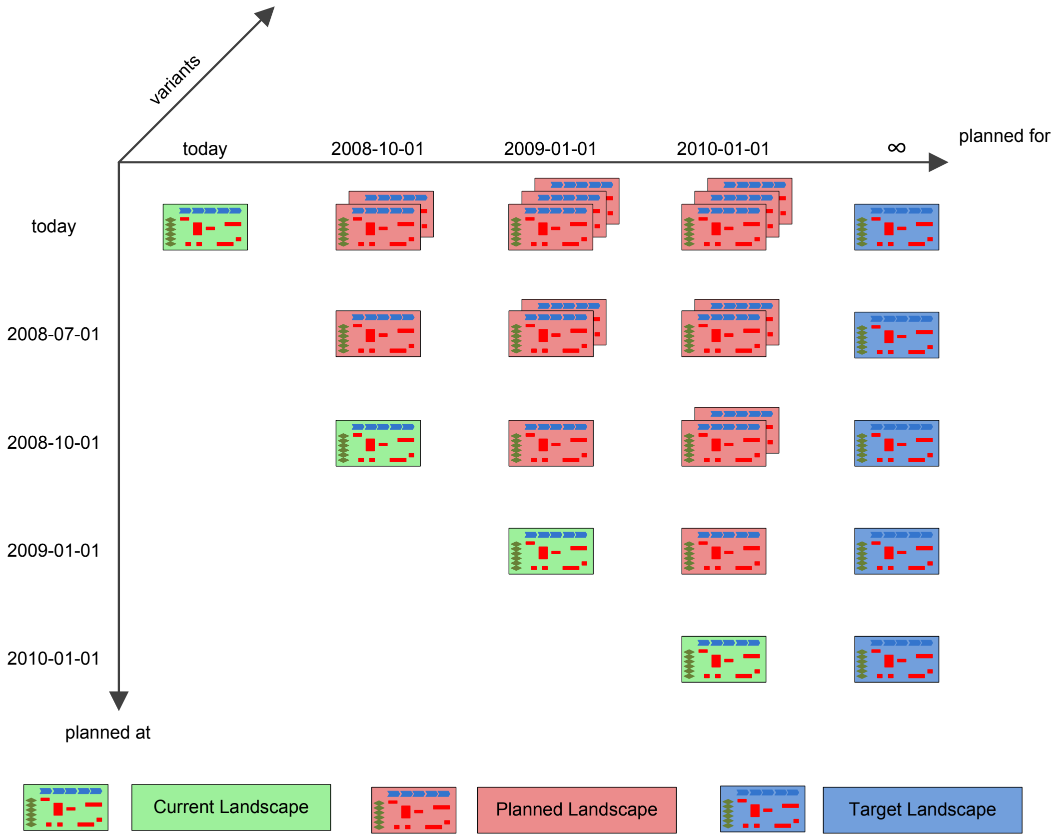


Figure 4.15.: Relationship of current, planned, and target landscape

Concerning aspects of changing landscapes, three different dimensions have to be taken into consideration as

- firstly, a landscape is planned for a specific time, as
- secondly, a landscape has been modeled at a certain time, and as
- thirdly, different variants of a planned landscape for a specific time may exist.

4.2.1. Landscape Management

The scenario *Landscape Management* simulates tasks relevant to planning and controlling the evolution of the application landscape and is concerned with the cross function *Demands & Projects* and the layers *Business* and *Application System* of the information model (see Section 3.2). The concerns of this scenario are:

> Information about the application landscape should be stored in the tool. Starting with the information about the current landscape potential development variants should be modeled. The information about the current application landscape and future states should be historicized in the tool to enable comparisons.
>
> Chosen versions of the application landscape, e. g. current, planned, and target landscapes should be analyzed and compared using different visualizations.

The above concerns especially emphasize on the three different dimensions as employed in landscape management:

The first two dimensions refer to temporal aspects of the application landscape. Thereby, the first time dimension expresses, when the planned landscape is prospected to be the current one, the second dimension of time shows, when the planning has taken place. In contrast, the third dimension represents the ability to model different landscape variants for the same specific date in the future.

In addressing the above concern questions as the following have to be answered:

- What does the *current* application landscape look like today?
- What is, according to the current *plan*, the application landscape going to look like in January 2008?
- What was, according to the *plan* of 01-01-2007, the application landscape going to look like in January 2008?
- What does the *target* application landscape do look like?
- What are the differences between the current landscape and the planned landscape, according to the current plan?
- What are the differences between the planned landscape according to the plan of 01-01-2007 and the current plan?

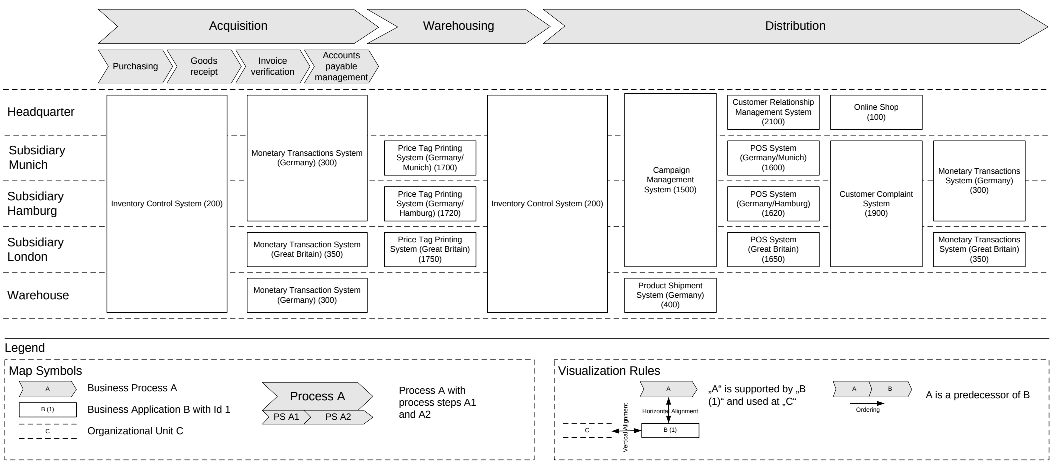


Figure 4.16.: Scenario *Landscape Management*: Current landscape

- What are the reasons for the differences between the current and the planned landscape, according to the current plan?
- What projects have to be initiated in order to change from the planned landscape (according to the current plan) to the target landscape? What planning scenarios can be envisioned and how do they look like?

The first deliverable is a graphical model of the current application landscape visualized as a process support map (see Figure 4.16).

The second deliverable shows the currently planned application landscape, where the project *Consolidation of Monetary Transaction Systems* is going to retire the application *Monetary Transaction System (Great Britain)* and the application *Monetary Transaction System (Germany)* is planed to take over its functionality. Figure 4.17 visualizes the currently planned application landscape, highlighting the differences from the current landscape.

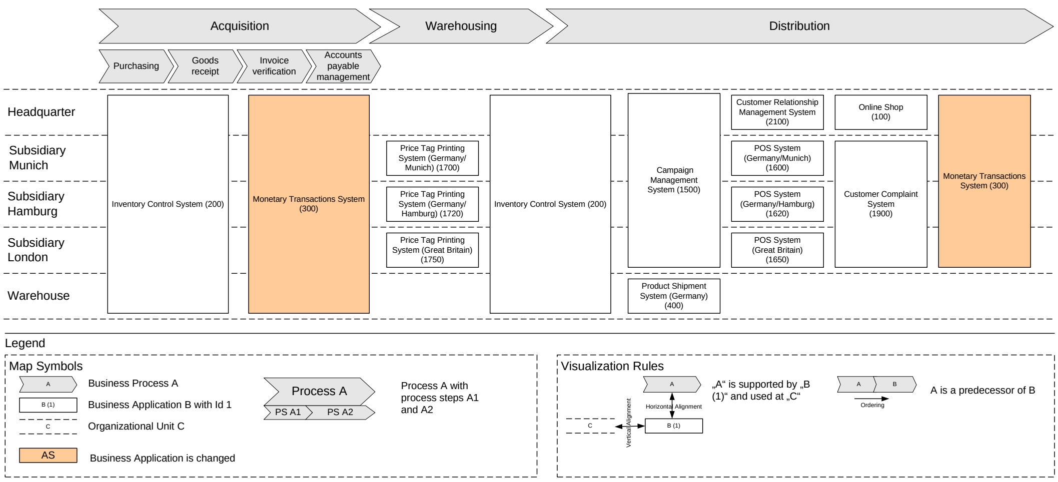


Figure 4.17.: Scenario *Landscape Management*: Planned landscape as of today

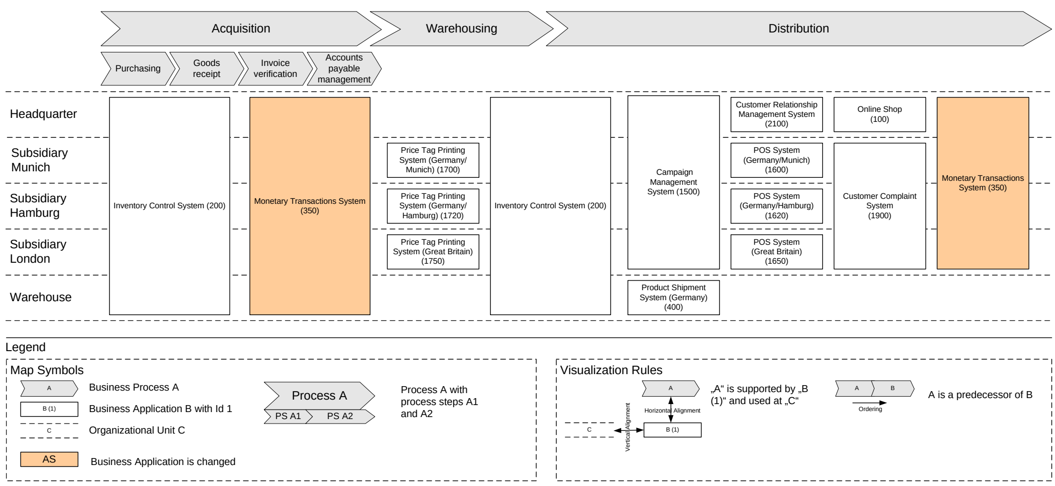


Figure 4.18.: Scenario *Landscape Management*: Planned landscape as of 01-01-2007

A third deliverable displays the planned application landscape as of 01-01-2007, where the project *Consolidation of Monetary Transaction Systems* was going to retire the application *Monetary Transaction System (Germany)* and the *Monetary Transaction System (Great Britain)* was planned to take over its functionality. Figure 4.18 visualizes the currently planned application landscape, highlighting the differences from the current landscape.

Finally, the target landscape developed by *SoCaStore*'s architects, which want to increase the vertical integration[4] over the different subsidiaries, represents the forth deliverable (see Figure 4.19). In the target landscape the *Price Tag Printing Systems* and the *POS Systems* are unified. The report (see Figure 4.20), as fifth deliverable, summarizes the differences between the current, the currently planned, and the target landscape.

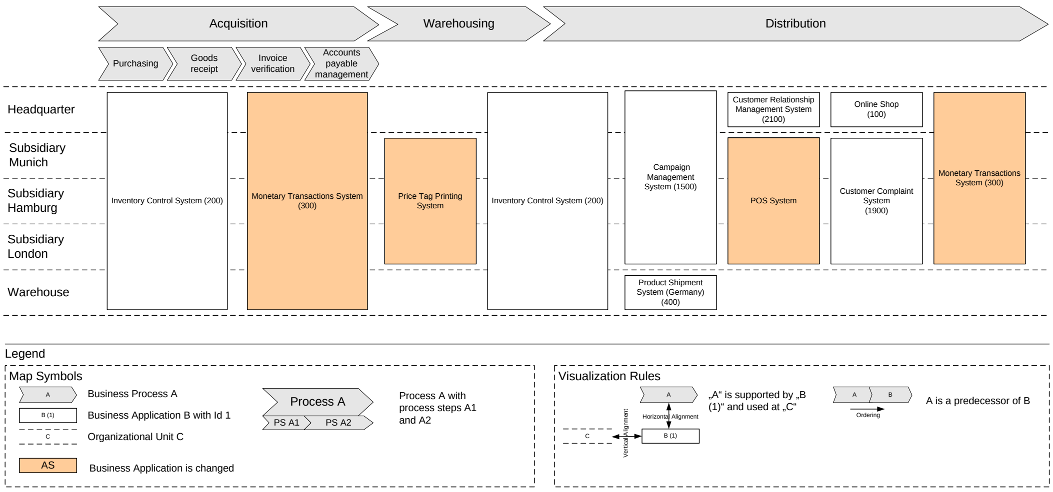


Figure 4.19.: Scenario *Landscape Management*: Target landscape

id	name	current	planned	target
		2008-01-01	2009-01-01	
100	Online Shop			
200	Inventory Control System			
300	Monetary Transactions System (Germany)		x	x
350	Monetary Transactions System (Great Britain)		x	x
400	Product Shipment System (Germany)			
1500	Campaign Management (Marketing Automation System)			
1600	POS System (Germany/Munich)			x
1620	POS System (Germany/Hamburg)			x
1650	POS System (Great Britain)			x
1700	Price Tag Printing System (Germany/Munich)			x
1720	Price Tag Printing System (Germany/Hamburg)			x
1750	Price Tag Printing System (Great Britain)			x
1900	Customer Complaint System			
2100	Customer Relationship Management System			

x - changed

Figure 4.20.: Scenario *Landscape Management*: Report for current, planned, and target landscape

[4]In this context an increase of the vertical integration means that a application system is used by more organizational units to support a specific process; increase of horizontal integration means that a application system supports more processes at a specific organizational unit.

4.2.2. Demand Management

The scenario *Demand Management* simulates the management process concerned with gathering and documenting the demands originating from business and IT. Thereby, one or more demands may result in one ore more IT projects.

This process supports the documentation of demands and linking them to the affected elements of the EA. Within the process, different demands, which require a similar functionality or which are related to the same application systems can be combined to one project proposal. This scenario addresses the cross function *Demands & Projects* while being likely to affect artifacts on all different layers of the information model (see Section 3.2). The concerns of this scenario are:

> The IT department of the *SoCaStore* department store has received numerous demands, which must be documented and linked to the affected elements of the EA. To prepare the project portfolio management a subset of the given demands has to be selected. These demands must subsequently be transformed into project proposals, combining demands asking for similar functionality or affecting the same application systems.

To address the concerns given above, the following questions should be answered:

- Which demands have been received?
- Which application systems are affected by the individual demands?
- Which demands can be combined into one project proposal?
- What are the resulting project proposals?

The information needed for this scenario are the different demands, including attributes like description, contact person, urgency, expected benefit, the support for the defined strategies and goals as well as the affected application systems.

As deliverable a report should be created that shows the documented demands and associated information as shown in Figure 4.21.

Id	Demands	Description	Contact person	Urgency	Affected Business Applications
1	Improve Customer retention	The percentage of customer relationships that a business is able to maintain on a long-term basis	Mr. Maier	3	100, 800, 1600, 1620, 1650, 2100
2	Homogenization	Reducing the heterogeneity of business applications	Mrs. Huber	1	100, 300, 350, 400, 700, 1000, 1100, 1200, 1400, 1500, 1600, 1620, 1650, 1720, 1750, 1820, 1850
3	Reduce costs	Cost-efficient support and maintenance for business applications	Mr. Hofer	2	100, 200, 300, 350, 400, 500, 700, 1000, 1100, 1200, 1400, 1500, 1600, 1620, 1650, 1720, 1750, 1820, 1850
4	Adapt to government regulations	Understanding the government regulations for business applications	Mr. Schmidt	4	100, 500, 600, 1600, 1620, 1650, 1700, 1720, 1750
5	Improve enterprise-wide knowledge management	Building an intellectual capital strategy	Mrs. Huber	2	800, 1100, 1200, 1300, 1900, 2000

Figure 4.21.: Scenario *Demand Management*: Report showing gathered demands and their attributes

For the definition of the resulting project proposals, an impact analysis shall be performed identifying demands longing for similar functionality or affecting the same application system, to identify candidates for combination. It should depict,

- which application systems are affected by different demands, and
- which demands ask for similar functionality.

To simplify the identification of demands that can be combined in one project proposal Figure 4.22 represents a second deliverable, which visualizes demands and the affected application systems.

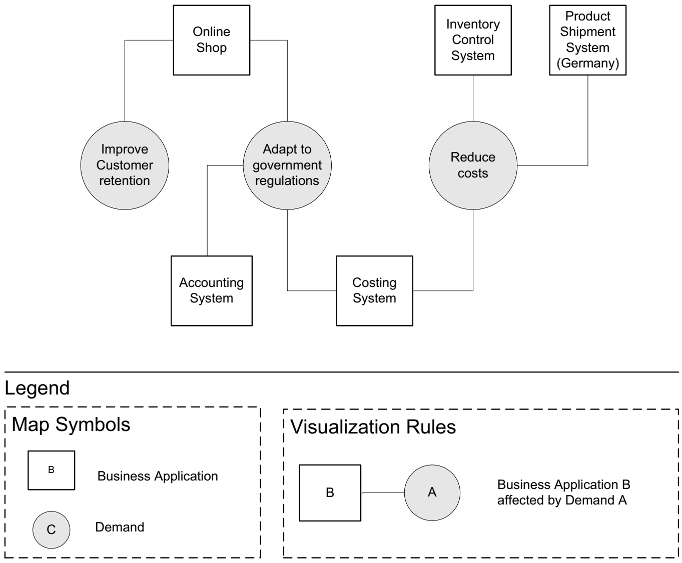


Figure 4.22.: Scenario *Demand Management*: Graphlayout map showing demands and the affected application systems

4.2.3. Project Portfolio Management

The scenario *Project Portfolio Management* simulates the management process concerned with retrieving the project portfolio for the next planning period.

This process supports the documentation of project proposals and linking the proposals with the affected elements of the EA. Measures like project costs, economic impact, etc. have to be documented in order to conduct different kinds of analysis on the portfolio, finally selecting a project portfolio to be realized. This scenario addresses the cross function *Demands & Projects* while being likely to affect artifacts on all different layers of the information model (see Section 3.2). The concerns of this scenario are:

> The IT department of the *SoCaStore* department store has received numerous project proposals. In consideration of the processes, organizational units, and application systems affected by the project proposals, a selection of the project proposals should be made. The available budget for projects is 5 million EUR.

The questions, which should be answered, are:

- Which project proposals have been received?
- What costs are calculated for which project proposal?
- What is the expected return of investment of which project proposal?
- Which application systems are modified/created/retired by the individual project proposals?
- Which processes/organizational units are affected by the changes as they use the application systems modified by the project proposals?
- Which project proposals have to be accomplished in any case?

The information needed for this scenario are the different projects proposals, including attributes like urgency, benefit, costs, expected return of investment, start date, and end date, as well as information about the criticality and size of application systems.

An impact analysis shall be performed transitively traversing the relationships between application systems, business processes, and organizational units. It should depict,

- which organizational units are affected by a project proposal as they use an application system affected by it, and
- which business processes are affected by a project proposal as they are supported by application systems modified by it.

As deliverable, reports should be created that show possible conflicts between project proposals arising from the concurrent modification of the same application (see Figure 4.23), showing business processes (see Figure 4.24), and showing organizational units (see Figure 4.25) transitively affected by project proposals. Finally, the proposal affecting certain business supports should be visualized (see Figure 4.26).

	Project id	1	5	6	7	8	9	10	11	12	13	14	15	16	17	18	20
Project id	**Project name**																
1	Integration of an auctioning platform into the online shop		x				x				x						
5	Database Consolidation			x			x	x	x		x	x	x	x	x	x	x
6	Consolidation of Monetary Transaction Systems							x						x	x		
7	VAT change in accounting system									x		x					
8	VAT change in costing system									x							
9	VAT change in online Shop										x						
10	VAT change in POS System													x			
11	VAT change in Price Tag Printing System																
12	Connection of Costing and Accounting System											x					
13	Reliability Improvement of the Online Shop																
14	Introduction of RFID in the Warehouse																
15	Replacement of the Customer Satisfaction Analysis System by the Data Warehouse													x			x
16	Introduction of a Bonus Card for Customers																x
17	Optimization of Monetary Transaction System																
18	Introduction of a new Management Information System																
20	Introduction of Knowledge Management																

x conflict

Figure 4.23.: Scenario *Project Portfolio Management*: Report showing possible conflicts of project proposals

		Process name (id)					
Project id	**Project name**	**Acquisition (1000)**	**Warehousing (3000)**	**Distribution (5000)**	**Human Resources (7000)**	**Controlling (8000)**	**Strategy and planning (9000)**
1	Integration of an auctioning platform into the online shop			x			
5	Database Consolidation	x	x	x	x	x	x
6	Consolidation of Monetary Transaction Systems	x		x	x		
7	VAT change in accounting system					x	
8	VAT change in costing system					x	
9	VAT change in online Shop			x			
10	VAT change in POS System			x			
11	VAT change in Price Tag Printing System		x				
12	Connection of Costing and Accounting System					x	
13	Reliability Improvement of the Online Shop			x			
14	Introduction of RFID in the Warehouse	x	x	x		x	
15	Replacement of the Customer Satisfaction Analysis System by the Data Warehouse			x		x	
16	Introduction of a Bonus Card for Customers			x		x	
17	Optimization of Monetary Transaction System	x		x			
18	Introduction of a new Management Information System				x	x	x
20	Introduction of Knowledge Management					x	

Figure 4.24.: Scenario *Project Portfolio Management*: Report showing, which project proposal affects which business process

Since financial aspects are of high interest for the *SoCaStore* department store another portfolio matrix shall be created visualizing the following: project proposals as bubbles, project proposal costs as the bubble radius, expected return of investment (ROI) rating on the y-axis, urgency on the x-axis, and

Project id	Project name	Organizational Units (id) Headquarter (1)	Subsidary Munich (2)	Subsidary Hamburg (3)	Subsidary London (4)	Warehouse (5)
1	Integration of an auctioning platform into the online shop	x				
5	Database Consolidation	x	x	x	x	x
6	Consolidation of "Monetary Transaction System"	x	x	x	x	x
7	VAT change in accounting system	x				
8	VAT change in costing system	x	x	x	x	x
9	VAT change in online Shop	x				
10	VAT change in POS System		x	x	x	
11	VAT change in Price Tag Printing System		x	x	x	
12	Connection of Costing and Accounting System	x	x	x	x	x
13	Reliability Improvement of the Online Shop	x				
14	Introduction of RFID in the Warehouse	x	x	x	x	x
15	Replacement of the Customer Satisfaction Analysis System by the Data Warehouse	x	x	x	x	
16	Introduction of a Bonus Card for Customers	x	x	x	x	
17	Optimization of Monetary Transaction System	x	x	x	x	x
18	Introduction of a new Management Information System	x				
20	Introduction of Knowledge Management	x				

Figure 4.25.: Scenario *Project Portfolio Management*: Report showing, which project proposals affects which organizational units

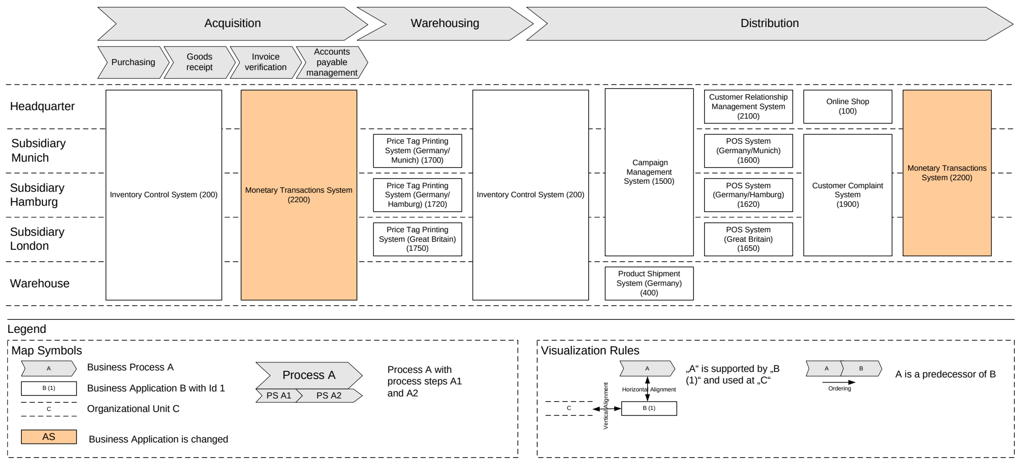


Figure 4.26.: Scenario *Project Portfolio Management*: Process support map showing, which project proposals affect which business support

the strategic impact rating shown as color coding on the project proposals. A possible deliverable is shown in Figure 4.27.

As several application systems are highly critical for the *SoCaStore* department store they should show a certain stability, which means that they should not be affected by too many projects. A *portfolio matrix* shall be created displaying the following information: application systems as bubbles, application system size as the bubble radius, application system criticality rating on the y-axis, number of project proposals affecting the application systems on the x-axis. The visualization might look like Figure 4.28.

4.2.4. Synchronization Management

The scenario *Synchronization Management* addresses issues of synchronizing projects, according to their interdependencies derived from the objects (e. g. application systems, services) a given project is likely to change. In this case, two projects that modify the same application system at the same time exert a conflict that is deemed worth avoiding. Thus, this scenario is concerned with the cross-function

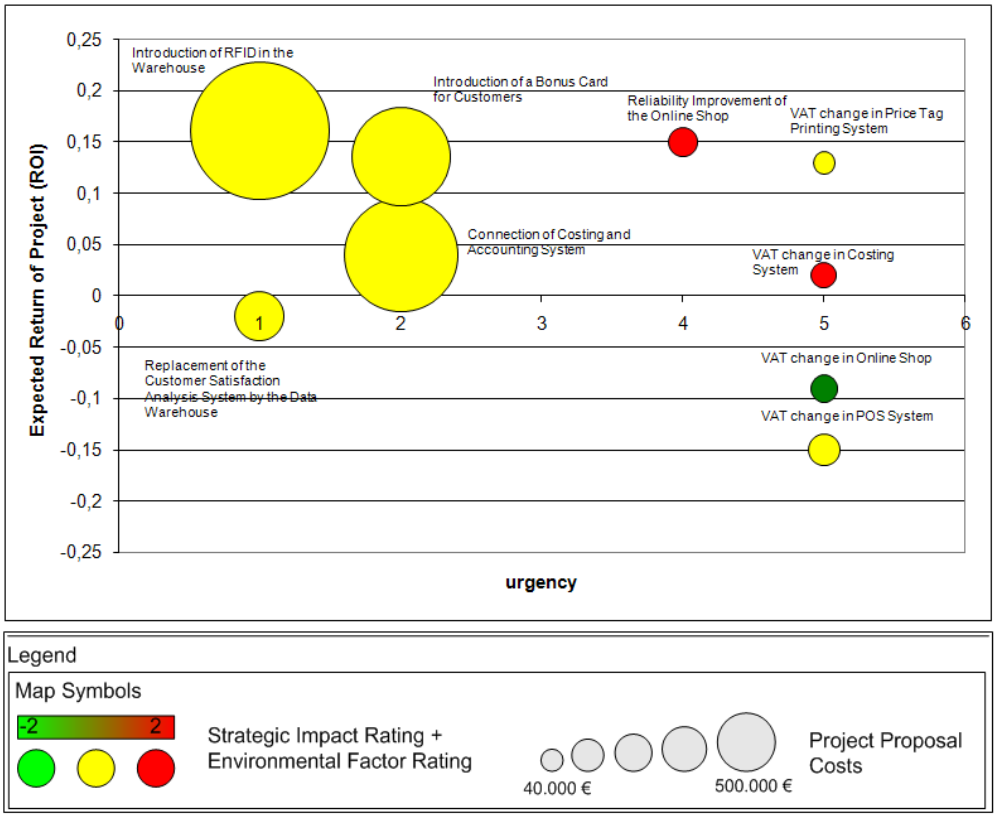


Figure 4.27.: Scenario *Project Portfolio Management*: Portfolio matrix showing the project proposals with the expected ROI, the urgency, and their strategic impact

Demands & Projects and all layers, as entities on every layer may be subject to changes executed in a project (see section 3.2). The concerns of this scenario are described as follows:

> To support the management of ongoing projects and to plan future projects, there has to be the possibility to model and manage project interdependencies or to derive them from the affected elements of the EA.
>
> It should be possible to analyze the project timeline using Gantt-like diagrams. This timeline shall than be updated and annotated to reflect delays of a single project as well as to identify projects, that depend on it and might also be delayed.

The questions are:

- Which projects affect the same organizational unit?
- Which dependencies exist among projects?
- What happens, if a particular project is delayed? Which schedules have to be adapted and how?

The information needed to accomplish this scenario are the projects to be executed with their start and end time, as well as their relationships to the application systems they affect. Analyzing relationships transitively, a deliverable shall be created showing, which organizational units are affected by which projects. The data needed can be derived from the already imported data by traversing the repository

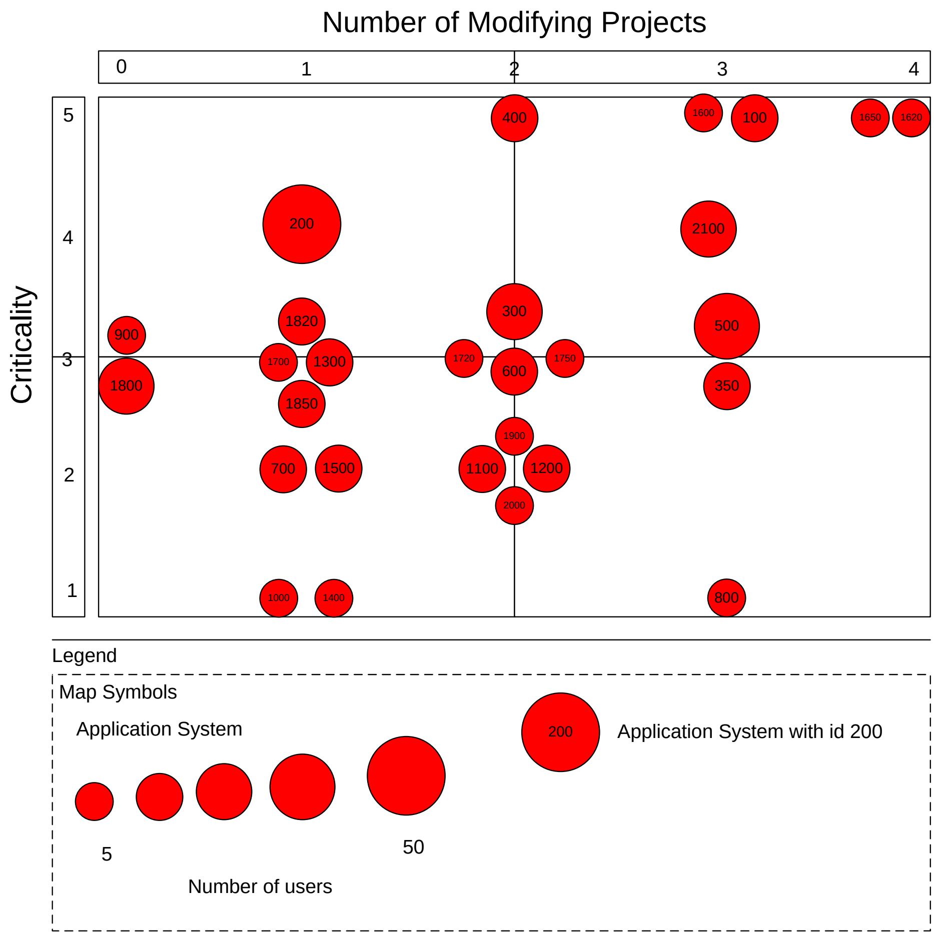


Figure 4.28.: Scenario *Project Portfolio Management*: Portfolio matrix showing the affected application systems with their criticality and number of modifying project proposals

from projects to affected application systems and from there to the organizational units, where these application systems are *used at*. A possible deliverable might look like Figure 4.30.

id	name	planned start	planned end	delayed start	delayed end
9	VAT change in online shop	15.09.2008	01.10.2008	01.10.2008	15.10.2008
13	Reliability Improvement of the Online Shop	01.10.2008	31.10.2008	15.10.2008	15.11.2008
16	Introduction of a bonus card for customers	01.11.2008	01.12.2008	15.11.2008	15.12.2008

Figure 4.29.: Scenario *Synchronization Management*: Report of the projects affected by a project delay

Further, a project delay shall be simulated, causing the project *Database consolidation* to end two weeks late. A visualization shall be created displaying the projects indirectly affected by this delay under the assumption that no application system should be modified by two projects at the same time. Such a deliverable could look like Figure 4.31. A further deliverable, a tabular report, containing information about the delay shall be created (see Figure 4.29).

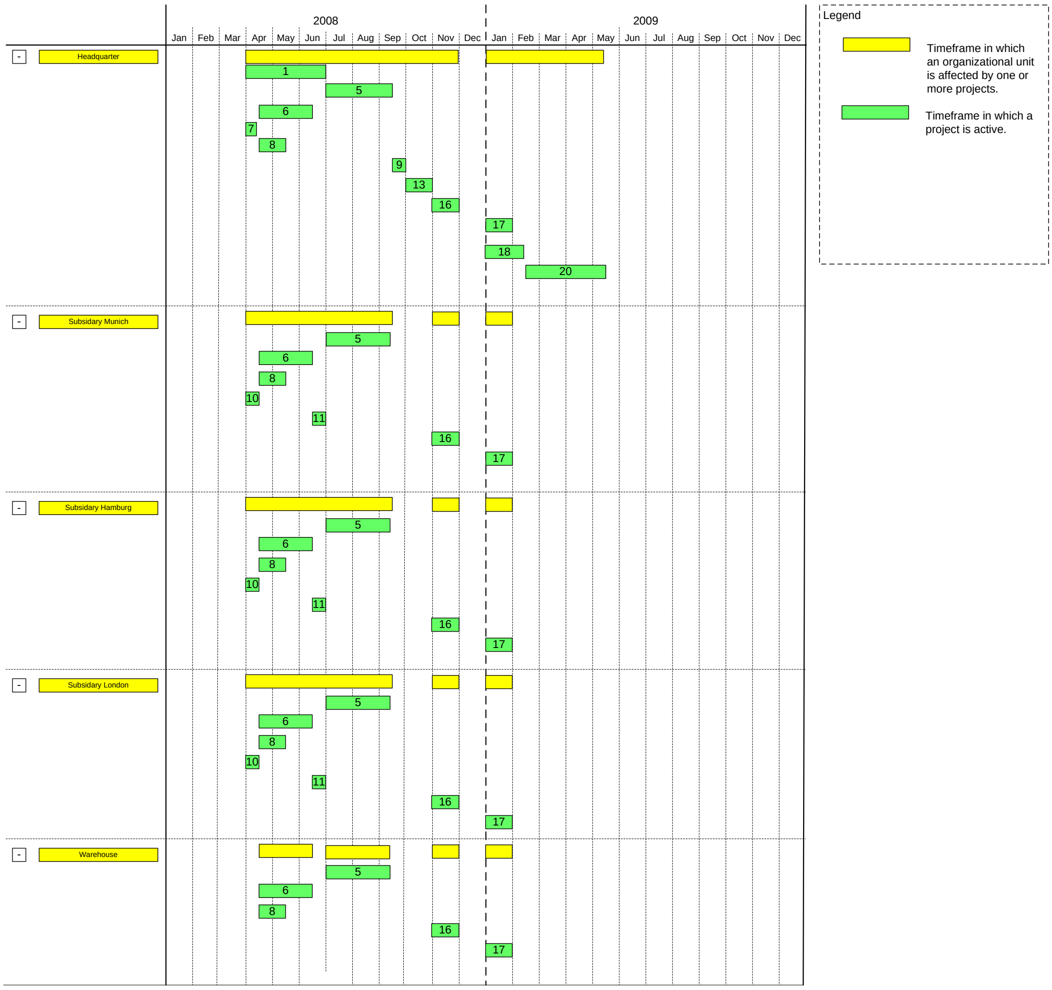


Figure 4.30.: Scenario *Synchronization Management*: Time interval map connecting projects to affected organizational units

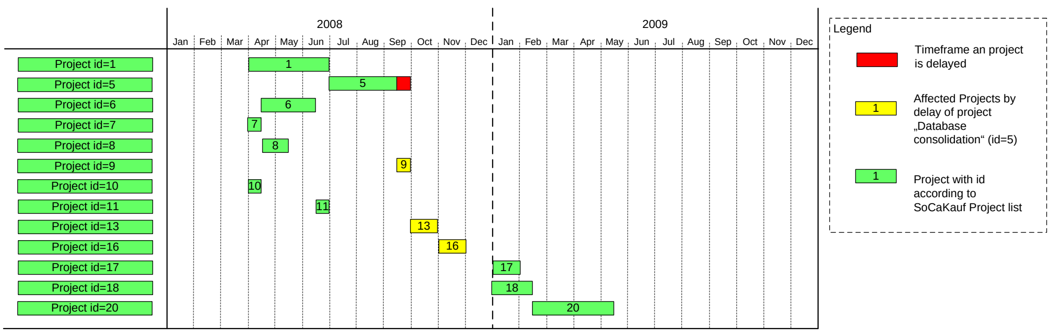


Figure 4.31.: Scenario *Synchronization Management*: Time interval map displaying projects affected indirectly by the delay of another project

The last management task to be accomplished here, is an impact analysis showing projects that are affected by the retirement of the application system *Customer Satisfaction Analysis System* to a given date. A visualization of this information might be achieved by color-coding the affected projects in a visualization like Figure 4.32.

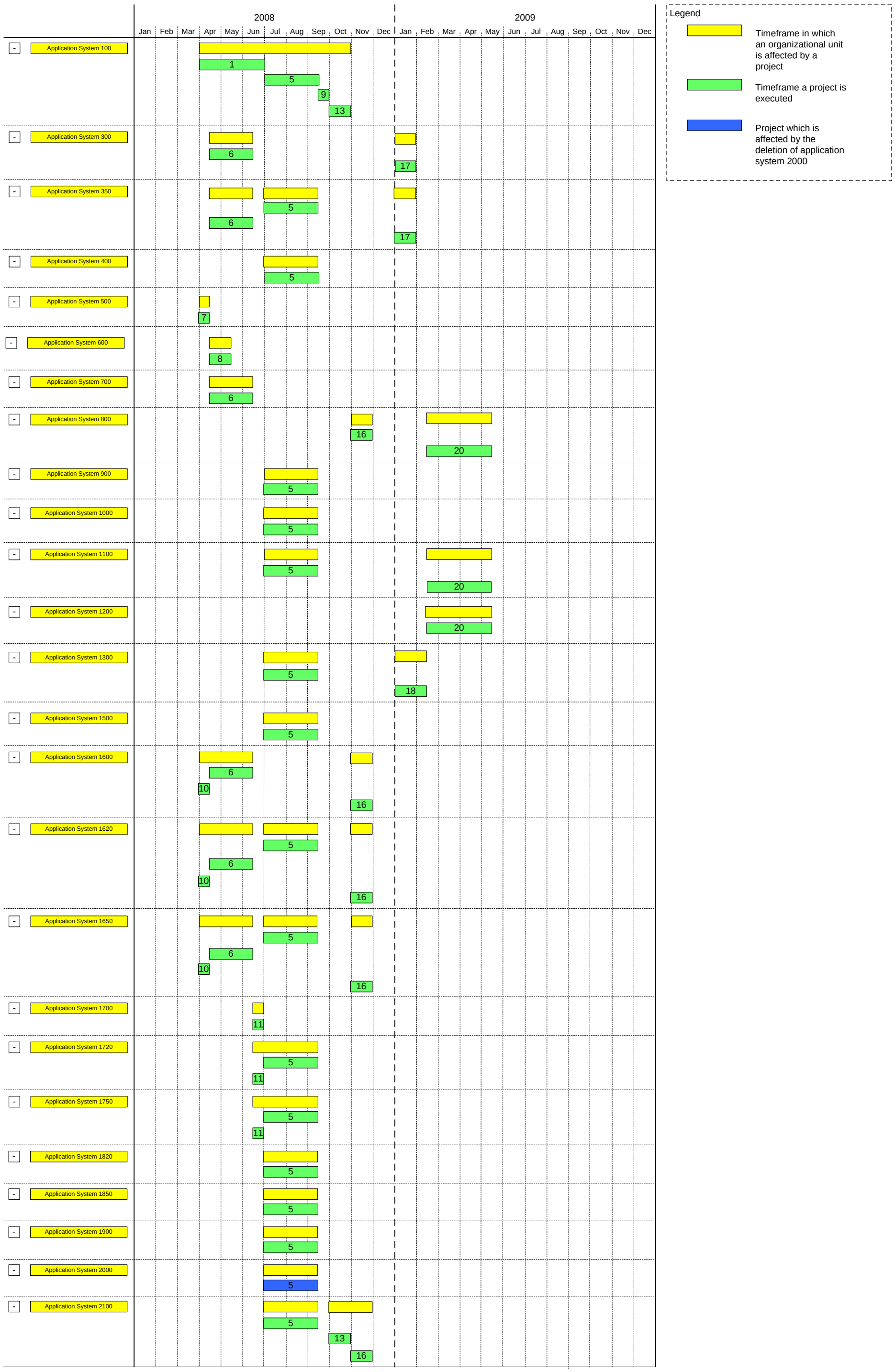


Figure 4.32.: Scenario *Synchronization Management*: Time interval map displaying projects affected by the retirement of an application

4.2.5. Strategies and Goals Management

The scenario *Strategies and Goals Management* addresses issues of aligning the EA management activities to the organization's strategies and goals. It is analyzed whether the tool supports transferring a strategy through the organizational hierarchy by decomposing it into smaller and more detailed pieces. This also includes tracing back the decomposition process, making it possible to trace from a specific action item on a fine grained level of the organizational hierarchy to the strategy it has been derived from. The cross function that is concerned here is *Strategies & Goals* according to the information model (see Section 3.2). All layers may be affected by this scenario. The concerns of this scenario are described as follows:

> As part of the implementation of a balanced scorecard at the *SoCaStore* department store the *customer perspective* is considered. The strategies and goals lead to different projects and changes in the EA. These changes should be traceable to the previously defined strategies and goals.

The questions derived from this concerns are:

- Which strategy leads to which goals?
- Have all goals been reached?
- Which organizational units have not reached their goals?
- Which projects support which goals?
- Which demands are related to which goals?

The information needed to accomplish this scenario consists of data on the enterprise's strategies and goals as well as on the relationships between the objects and to the projects planned and executed. Further, information about metrics operationalizing the goals should be added.

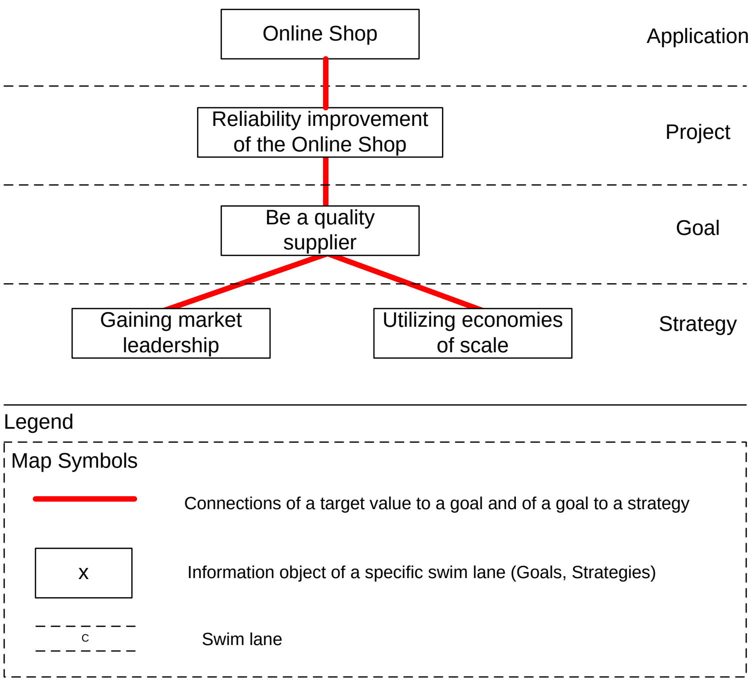


Figure 4.33.: Scenario *Strategies and Goals Management*: Impact analysis starting with a project

A deliverable tracing from a project up to the related strategies, traversing the affected application systems and demands, is shown in Figure 4.33. Figure 4.34 represents a deliverable visualizing the hierarchic trace of a strategy down to the implementing projects.

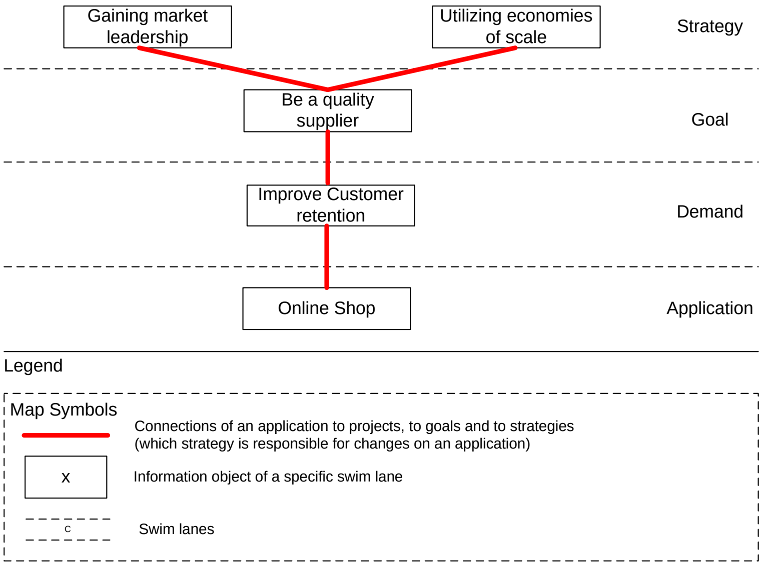


Figure 4.34.: Scenario *Strategies and Goals Management*: Impact analysis starting with a strategy

A tabular report (see Figure 4.35) should show the goals contributing to pursuing a specific strategy. The fulfillment of a specific goal up to the current date can be derived from a corresponding metric operationalizing the goal and target values defined for them. Showing the current state of a goal's fulfillment is also very important in keeping track of the enterprises' strategies and goals. A tabular report containing this information is given in figure 4.36.

	Goal						
Strategy	**Be cost effective (X1)**	**Be a quality supplier (X2)**	**Reduce storage time (A1)**	**Reduce time to delivery (B1)**	**Increase number of purchases (C1)**	**Reduce complaints (D1)**	**Increase customer satisfaction (D2)**
Gaining market leadership (1)		x	x	x	x	x	x
Utilizing economy of scale (2)	x	x	x	x	x		

Figure 4.35.: Scenario *Strategies and Goals Management*: Report with strategies and goals

goal	organizational unit	metric	2008-05-01	2008-08-01	target	fullfillment	indicator
Be cost effective (X1)	SoKaKauf	Sum of personel, material and capital cost	50000000	42001471	45000000	159,97%	
Be a quality supplier (X2)	SoKaKauf	Average customer satisfaction according to a performed survey (2)	6	7	7	100,00%	
Reduce storage time (A1)	Warehouse (5)	Average time in storage (for an article) in days (3)	10	8	9	200,00%	
Reduce time to delivery (B1)	Warehouse (5)	Maximum delivery time in days (4)	30	25	21	55,56%	
		Average delivery time in days (5)	5	5	4,5	0,00%	
Increase number of purchases (C1)	Warehouse (5)	Number of purchases per day (6)	14000	15000	15400	71,43%	
Reduce Complaints (D1)	Subsidary Munich (2)	Number of complaints per day (7)	10	9	8	50,00%	
Increase customer satisfaction (D2)	Subsidary Munich (2)	Average customer satisfaction according to a performed survey (8)	6	7	7	100,00%	

Figure 4.36.: Scenario *Strategies and Goals Management*: Report with goals, metrics, and measures

4.2.6. Business Object Management

This scenario simulates the tools capabilities to model business objects as well as the operations performed on them and the exchange of the business objects between the application systems. In this scenario mainly the layer *Business Service* is concerned with some additional links to the layers *Business* and *Application System*. The main concerns of the scenario are described as follows:

> The department store *SoCaStore* wants to get an overview of the business objects involved and exchanged in the execution of the business processes. Therein, especially the data flow between the application systems performing operations on the business objects should be modeled and the kind of operation performed in a specific application system should be detailed. Furthermore, the technical realizations of the data flows should be taken into consideration, as should exchange steps between application systems, where additional manual operations have to take place.

The questions in this context are:

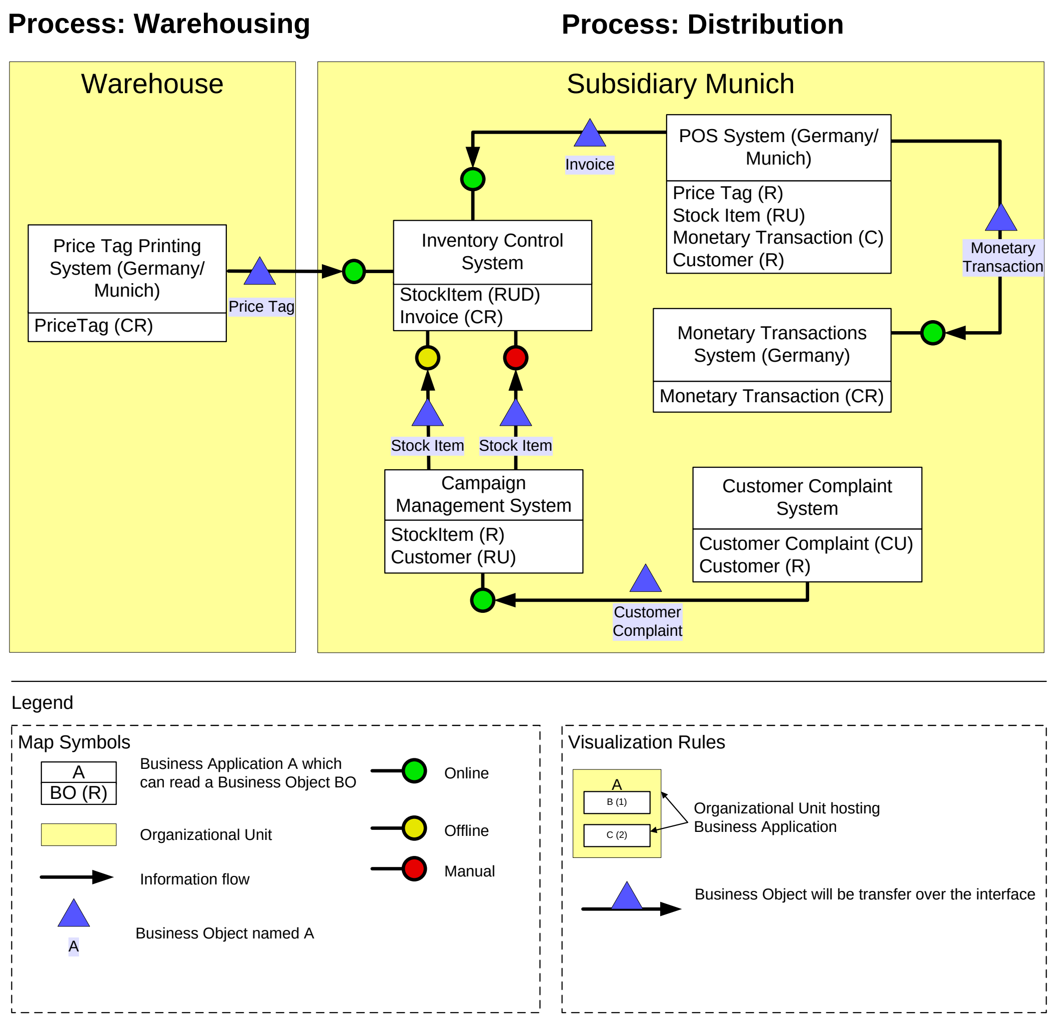


Figure 4.37.: Scenario *Business Object Management*: Cluster map showing information flows between application systems

- Which business objects are created, modified, or deleted by which application systems during the execution of which business process?
- Which application system exchange which business objects via which technical interface?
- Which application system holds the master copy of which business object?
- Which business processes need additional manual operations in their execution?

In simulating this scenario information about *Application Systems* offering *Business Services* for supporting *Business Processes* via *Connectors* is needed. Further, information about *Interconnections* linking application systems as well as the *Business Objects* exchanged over these links is required.

The information flow represented by exchanged business objects should be visualized for specific business processes at specific organizational units (for an exemplary deliverable see Figure 4.37). The connectors of the application systems may additionally provide information about the connection type.

A matrix and a report showing, which application systems communicate with each other and which business objects are exchanged (see deliverable in Figure 4.38) should be created.

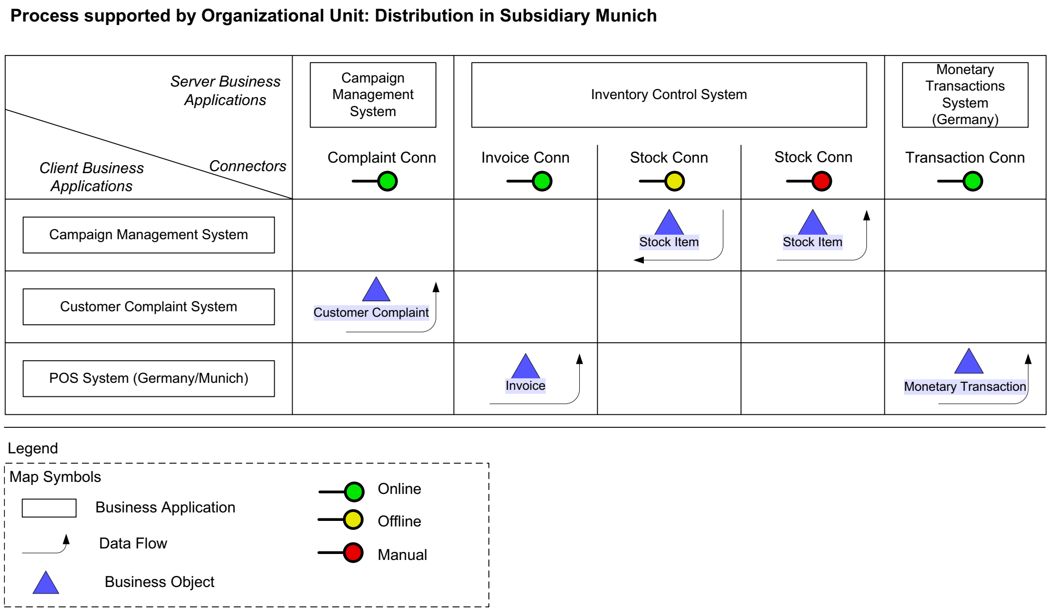


Figure 4.38.: Scenario *Business Object Management*: Matrix map showing information flows between application systems

4.2.7. SOA Transformation

This scenario simulates the tools capabilities to support the enterprise in transforming their architecture into a *Service Oriented Architecture (SOA)*. In this scenario mainly the layer *Business Service* is concerned with some additional links to the layers *Business* and *Application System*. The main concerns of the scenario are described as follows:

> The department store *SoCaStore* wants to transform its architecture into a service oriented one. Thereby, a top-down as well as a bottom-up approach for the identification of possible candidates for reusable services will be used. The top-down approach identifies services according to the usage of business objects within the conduction of different

business processes, whereas the bottom-up approach identifies technical functionalities currently provided by applications, which should be transformed into reusable services to demonstrate the benefit. Besides the identification of applicable candidates, the effects the transformation will have on the application landscape should be modeled and service level agreements for the individual services must be defined.

The questions in this context are:

- Does the application system support a differentiating or a standardized business process?
- Does the application system change frequently?
- Which application systems will be affected by a change in the near future?
- Which applications are used within numerous domains?
- Which operations are performed on business objects and how often are these operations used?
- What service level should a certain business service fulfill?
- Which application systems are changed during the transformation to a SOA?
- How are the changes reflected in the application landscape?

During the simulation of this scenario information about *Application Systems* offering *Functionalities* for supporting *Business Processes* as well as the number of usages within different domains is needed. Further, information about *Business Processes* their character, *Business Objects* and the operations performed on them, and their usage within the execution of *Business Processes* is required.

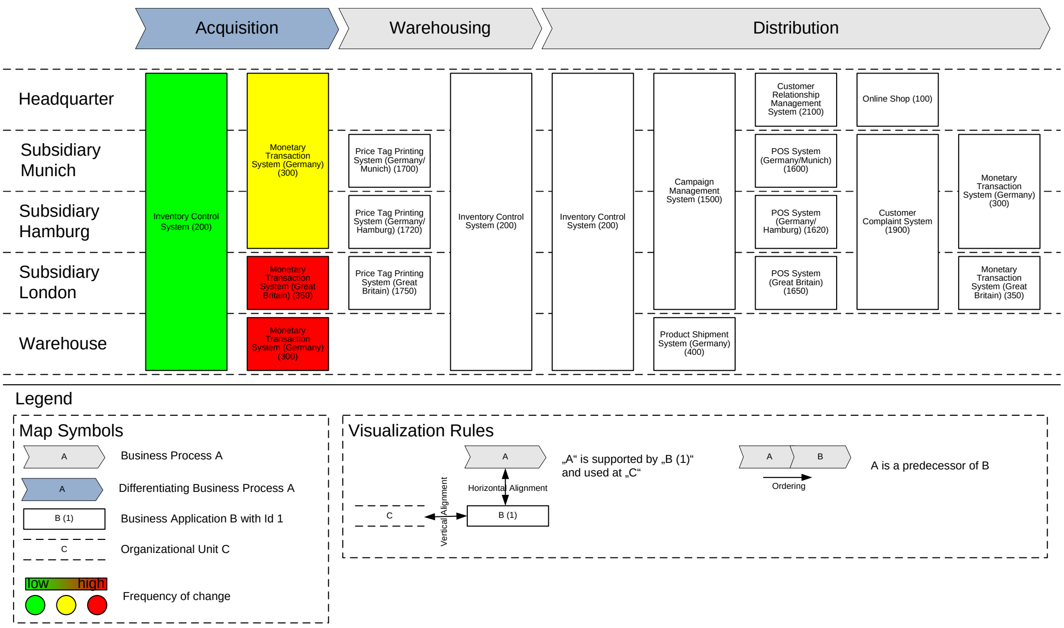


Figure 4.39.: Scenario *SOA Transformation*: Business process map showing differentiating business processes and supporting application systems color-coding

A first deliverable is a process support map, highlighting the differentiating business processes[5] and showing the supporting application systems color-coded by their frequency of change and the organizational units using them (see Figure 4.39). A report, as second deliverable, shows business objects

[5]Differencing business processes in this context are business processes, which are critical for the enterprise‘s unique selling position.

and the operations performed on them by application systems (see Figure 4.40). Further, a report itemizing the applications that will be changed within the next year, ordered by their date of change (see Figure 4.41), can be used to identify applicable candidates for the definition of a reusable service. Another report (see Figure 4.42) should be created, listing the future business services and their intended service level agreements.

Business Process: Distribution

		Price Tag	Stock Item	Monetary Transaction	Customer Complaint	Customer	Invoice
Headquarter	Inventory Control System (200)		RUD				CR
	Campaign Management System (1500)		R			RU	
	Customer Relationship Management System (2100)				CRUD	CRUD	
	Online Shop (100)		RU			CRU	
Subsidiary Munich	Inventory Control System (200)		RUD				CR
	Campaign Management System (1500)		R			R	
	POS System (Germany/Munich) (1600)	R	RU	C		R	
	Customer Complaint System (1900)				CU	R	
	Monetary Transaction System (Germany) (300)			CR			
Subsidiary Hamburg	Inventory Control System (200)		RUD				CR
	Campaign Management System (1500)		R			RU	
	POS System (Germany/Hamburg) (1620)	R	RU	C		R	
	Customer Complaint System (1900)				CU	R	
	Monetary Transaction System (Germany) (300)			CR			
Subsidiary London	Inventory Control System (200)		RUD				CR
	Campaign Management System (1500)		R			RU	
	POS System (Great Britain) (1650)	R	RU	C		R	
	Customer Complaint System (1900)				CU	R	
	Monetary Transaction System (Great Britain) (350)			CR			
Warehouse	Inventory Control System (200)		RUD				CR
	Product Shipment System (Germany) (400)		CRU	C			

C.. Create
R.. Read
U.. Update
D.. Delete

Figure 4.40.: Scenario *SOA Transformation*: Report showing business objects and the operations performed on them by application systems

id	name	productionFrom
1800	Worktime Management System (Germany/Munich)	01.01.2007
1820	Worktime Management System (Germany/Hamburg)	01.01.2007
1850	Worktime Management System (Great Britain)	01.01.2007
520	Accounting System	01.02.2007
1200	Supplier Relationship Management System	01.03.2007
1700	Price Tag Printing System (Germany/Munich)	01.03.2007
1720	Price Tag Printing System (Germany/Hamburg)	01.03.2007
1750	Price Tag Printing System (Great Britain)	01.03.2007
900	Fleet Management System	01.04.2007
1600	POS System (Germany/Munich)	01.04.2007
1620	POS System (Germany/Hamburg)	01.04.2007
1650	POS System (Great Britain)	01.04.2007
1900	Customer Complaint System	01.04.2007
500	Accounting System	01.05.2007
2100	Customer Relationship Management System	01.05.2007
700	Human Resources System	01.06.2007
1000	Business Traveling System	01.06.2007
100	Online Shop	01.07.2007
800	Data Warehouse	01.07.2007
300	Monetary Transactions System (Germany)	01.08.2007
1500	Campaign Management System	01.08.2007
600	Costing System	15.09.2007
200	Inventory Control System	01.10.2007
2020	Customer Satisfaction Analysis System	01.10.2007
1400	Financial Planning System	01.12.2007

Figure 4.41.: Scenario *SOA Transformation*: Change report showing the applications that will be changed within the next year

id	name	description	service level agreement
2500	Price Tag Printing Service	Service that provides price tag printing	7-19, Mo-Fr

Figure 4.42.: Scenario *SOA Transformation*: Report showing the future business services and their service level agreements

In order to get an overview about effects of the transformation a time interval map (Figure 4.43) should be created visualizing the relocation of functionalities from application systems to business services.

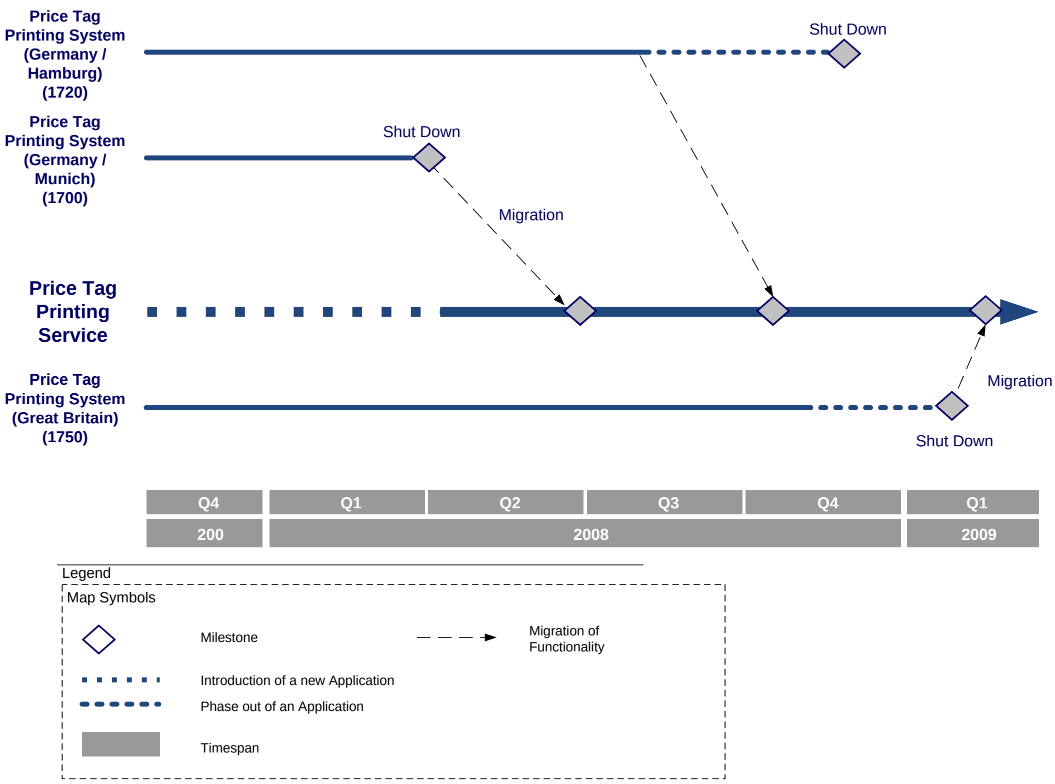


Figure 4.43.: Scenario *SOA Transformation*: Time interval map showing the migration of functionalities from an application system to business services

4.2.8. IT Architecture Management

The scenario *IT Architecture Management* deals with the introduction and the implementation of blueprints standardizing the architecture of specific application systems. This means introducing architectural standards, future application systems should be based on and current application systems should be adapted to. In terms of the information model outlined in Section 3.2, the affected cross function is *Guidelines & Patterns*, among the layers mainly *Application System* is affected. The concerns of this scenario are described as follows:

> *SoCaStore* regards its heterogeneous application landscape as a problem. The high number of technologies used in different architectures calls for a high number of experts. A homogenization may reduce operating costs, e. g. by consolidation of used software licenses, and maintenance expenses, e. g. by reducing administrative efforts.

The questions of relevance in this context are:

- Which architectural solutions are used within the different domains of *SoCaStore*?
- Which of the existing architectures should be maintained, which should be replaced?
- Which solution elements are used within *SoCaStore*?
- Which of the existing solution elements should be kept, which should be replaced?
- Which application systems use which solution elements?
- What actions have to be derived, if architectures/solution elements are replaced?

The information needed to accomplish this scenario consists of information about the *blueprint elements*, classes of technologies, such as "Webserver" or "Database", the *solution elements* realizing them, such as "Apache 2.0" or "Oracle 9.2i", the *architectural blueprints*, such as "4-tier-thin-client-architecture", making use of abstract technologies, and the *architectural solutions* realizing a specific architectural blueprint by utilizing technologies. Besides, it should be possible to define one or more criteria that are relevant due to an architectural solution, e. g. high availability or modularization.

A first deliverable is a cluster map, showing by color-coding which application system is compliant with which architectural solution (see Figure 4.44). Further, a report shall be created giving information about the relationships of the application systems to the architectural solutions they are based on and to the solution elements used to realize the given architectural solution (see deliverable in Figure 4.45).

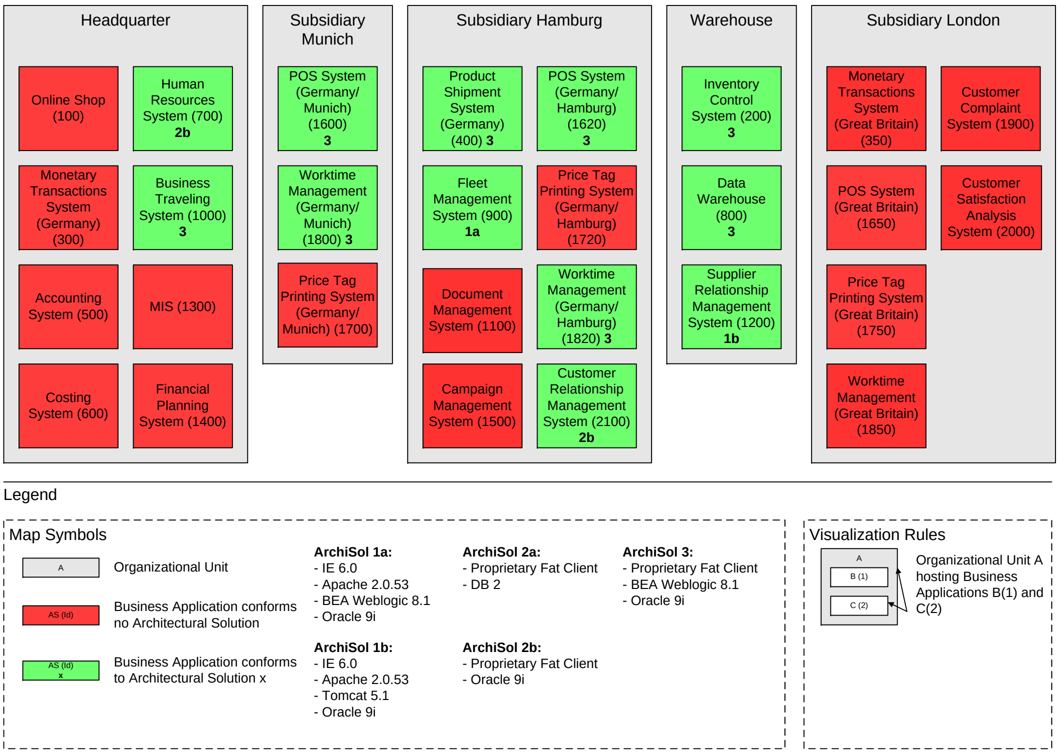


Figure 4.44.: Scenario *IT Architecture Management*: Cluster map showing application systems and their conformance with architectural solutions

id	name	architectural solution	Used Solution Elements	Used Architectural Blueprint
100	Online Shop 1.0	OracleTomcat	Tomcat 5.1, Apache 2.0.53, IE 6.0, Oracle 9i	4-Tier-Thin-Client-Architecture
120	Online Shop 1.5	OracleTomcat	Tomcat 5.1, Apache 2.0.53, IE 6.0, Oracle 9i	4-Tier-Thin-Client-Architecture
200	Inventory Control System 1.0			
220	Inventory Control System 1.5			
300	Monetary Transactions System (Germany)	DB2BeaFatClient	Bea Weblogic 8.1, Proprietary Fat-Client, DB2 6.0	3-Tier-Fat-Client-Architecture
350	Monetary Transactions System (Great Britain)	DB2BeaFatClient	Bea Weblogic 8.1, Proprietary Fat-Client, DB2 6.1	3-Tier-Fat-Client-Architecture
400	Product Shipment System (Germany)	MySQLTomcat	MySQL 2.1, Tomcat 5.1, Apache 2.0.53, IE 6.0	4-Tier-Thin-Client-Architecture
500	Accounting System 1.5	OracleTomcat	Tomcat 5.1, Apache 2.0.53, IE 6.0, Oracle 9i	4-Tier-Thin-Client-Architecture
510	Accounting System 2.0	OracleTomcat	Tomcat 5.1, Apache 2.0.53, IE 6.0, Oracle 9i	4-Tier-Thin-Client-Architecture
520	Accounting System 2.5	OracleTomcat	Tomcat 5.1, Apache 2.0.53, IE 6.0, Oracle 9i	4-Tier-Thin-Client-Architecture
600	Costing System	MySQLBeaFatClient	MySQL 2.1, Bea Weblogic 8.1, Proprietary Fat-Client	3-Tier-Fat-Client-Architecture
700	Human Resources System	OracleBea	Apache 2.0.53, IE 6.0, Oracle 9i, Bea Weblogic 8.1	4-Tier-Thin-Client-Architecture
800	Data Warehouse	OracleBea	Apache 2.0.53, IE 6.0, Oracle 9i, Bea Weblogic 8.2	4-Tier-Thin-Client-Architecture
900	Fleet Management System	MySQLTomcat	MySQL 2.1, Tomcat 5.1, Apache 2.0.53, IE 6.1	4-Tier-Thin-Client-Architecture
1000	Business Traveling System 1.0	BeaFatClient	Oracle 9i, Bea Weblogic 8.1, Proprietary Fat-Client	3-Tier-Fat-Client-Architecture
1020	Business Traveling System 1.5	BeaFatClient	Oracle 9i, Bea Weblogic 8.1, Proprietary Fat-Client	3-Tier-Fat-Client-Architecture
1100	Document Management System	OracleFatClient	Oracle 9i, Proprietary Fat-Client	2-Tier-Fat-Client-Architecture
1200	Supplier Relationship Management System	OracleTomcat	Tomcat 5.1, Apache 2.0.53, IE 6.0, Oracle 9i	4-Tier-Thin-Client-Architecture
1300	MIS (Management Information System) 1.0	MySQLTomcat	MySQL 2.1, Tomcat 5.1, Apache 2.0.53, IE 6.2	4-Tier-Thin-Client-Architecture
1320	MIS (Management Information System) 2.0	MySQLTomcat	MySQL 2.1, Tomcat 5.1, Apache 2.0.53, IE 6.3	4-Tier-Thin-Client-Architecture
1340	MIS (Management Information System) 3.0	MySQLTomcat	MySQL 2.1, Tomcat 5.1, Apache 2.0.53, IE 6.4	4-Tier-Thin-Client-Architecture
1400	Financial Planning System	OracleFatClient	Oracle 9i, Proprietary Fat-Client	2-Tier-Fat-Client-Architecture
1500	Campaign Management System	OracleTomcat	Tomcat 5.1, Apache 2.0.53, IE 6.0, Oracle 9i	4-Tier-Thin-Client-Architecture
1600	POS System (Germany/Munich) 1.0	OracleFatClient	Oracle 9i, Proprietary Fat-Client	2-Tier-Fat-Client-Architecture
1605	POS System (Germany/Munich) 1.5	OracleFatClient	Oracle 9i, Proprietary Fat-Client	2-Tier-Fat-Client-Architecture
1620	POS System (Germany/Hamburg) 1.0	BeaFatClient	Oracle 9i, Bea Weblogic 8.1, Proprietary Fat-Client	3-Tier-Fat-Client-Architecture
1625	POS System (Germany/Hamburg) 1.5	BeaFatClient	Oracle 9i, Bea Weblogic 8.1, Proprietary Fat-Client	3-Tier-Fat-Client-Architecture
1650	POS System (Great Britain) 1.0	BeaFatClient	Oracle 9i, Bea Weblogic 8.1, Proprietary Fat-Client	3-Tier-Fat-Client-Architecture
1655	POS System (Great Britain) 1.5	BeaFatClient	Oracle 9i, Bea Weblogic 8.1, Proprietary Fat-Client	3-Tier-Fat-Client-Architecture
1700	Price Tag Printing System (Germany/Munich) 1.0	MySQLBeaFatClient	MySQL 2.1, Bea Weblogic 8.1, Proprietary Fat-Client	3-Tier-Fat-Client-Architecture
1705	Price Tag Printing System (Germany/Munich) 1.5	MySQLBeaFatClient	MySQL 2.1, Bea Weblogic 8.1, Proprietary Fat-Client	3-Tier-Fat-Client-Architecture
1720	Price Tag Printing System (Germany/Hamburg) 1.0	MySQLBeaFatClient	MySQL 2.1, Bea Weblogic 8.1, Proprietary Fat-Client	3-Tier-Fat-Client-Architecture
1725	Price Tag Printing System (Germany/Hamburg) 1.5	MySQLBeaFatClient	MySQL 2.1, Bea Weblogic 8.1, Proprietary Fat-Client	3-Tier-Fat-Client-Architecture
1750	Price Tag Printing System (Great Britain) 1.0	MySQLBeaFatClient	MySQL 2.1, Bea Weblogic 8.1, Proprietary Fat-Client	3-Tier-Fat-Client-Architecture
1755	Price Tag Printing System (Great Britain) 1.5	MySQLBeaFatClient	MySQL 2.1, Bea Weblogic 8.1, Proprietary Fat-Client	3-Tier-Fat-Client-Architecture
1800	Worktime Management System (Germany/Munich)	OracleFatClient	Oracle 9i, Proprietary Fat-Client	2-Tier-Fat-Client-Architecture
1820	Worktime Management System (Germany/Hamburg)	OracleFatClient	Oracle 9i, Proprietary Fat-Client	2-Tier-Fat-Client-Architecture
1850	Worktime Management System (Great Britain)	OracleFatClient	Oracle 9i, Proprietary Fat-Client	2-Tier-Fat-Client-Architecture
1900	Customer Complaint System	OracleTomcat	Tomcat 5.1, Apache 2.0.53, IE 6.0, Oracle 9i	4-Tier-Thin-Client-Architecture
2000	Customer Satisfaction Analysis System 1.0	OracleTomcat	Tomcat 5.1, Apache 2.0.53, IE 6.0, Oracle 9i	4-Tier-Thin-Client-Architecture
2020	Customer Satisfaction Analysis System 1.2	OracleTomcat	Tomcat 5.1, Apache 2.0.53, IE 6.0, Oracle 9i	4-Tier-Thin-Client-Architecture
2100	Customer Relationship Management System			
2200	Monetary Transaction System	OracleFatClient	Oracle 9i, Proprietary Fat-Client	2-Tier-Fat-Client-Architecture
2300	Accounting and Costing System	OracleTomcat	Tomcat 5.1, Apache 2.0.53, IE 6.0, Oracle 9i	4-Tier-Thin-Client-Architecture
2400	Knowledge Managment System	OracleTomcat	Tomcat 5.1, Apache 2.0.53, IE 6.0, Oracle 9i	4-Tier-Thin-Client-Architecture

Figure 4.45.: Scenario *IT Architecture Management*: Report showing application systems and their conformance with architectural solutions

In order to get information about the degree of homogenization of the application landscape it is also of interest to analyze how often an architectural solution is used within the different domains of *SoCaStore*. Figure 4.46 visualizes the number of usage for each architectural solution.

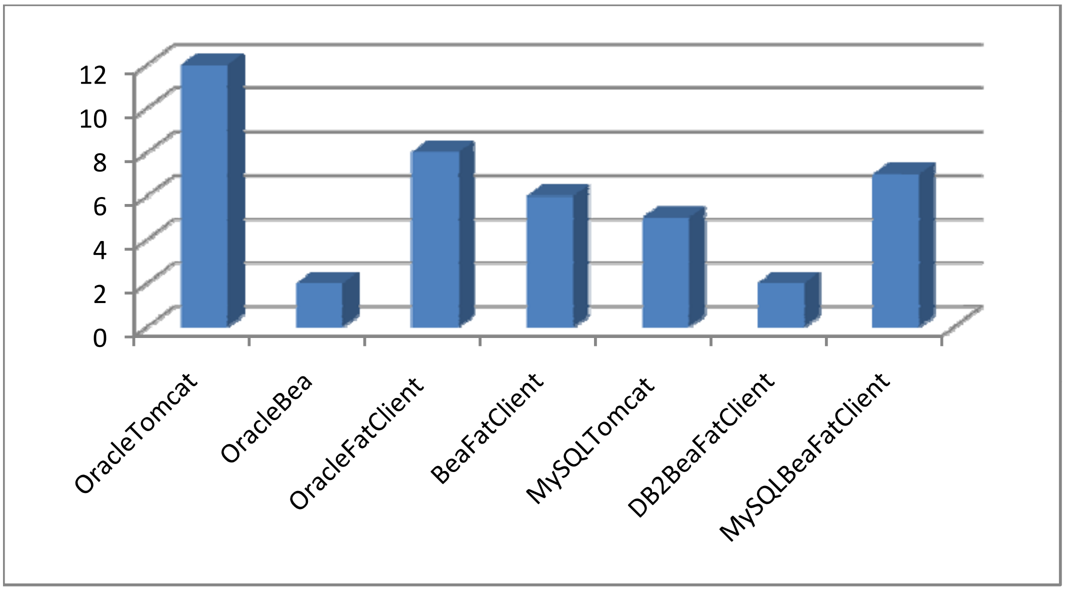


Figure 4.46.: Scenario *IT Architecture Management*: Report showing number of usage of architectural solutions

4.2.9. Infrastructure Management

The scenario *Infrastructure Management* deals with issues of the IT infrastructure of the enterprise. This infrastructure contains but is not limited to database and middleware systems. Thus, in the terms of the information model outlined in Section 3.2, the affected cross function is *Guidelines & Patterns*, the affected layers are mainly *Infrastructure*, *Application System*, and *Business*. The concerns of this scenario are described as follows:

> The department store *SoCaStore* intends to consolidate its database systems to decrease the costs for maintenance and licences. Also, expected support periods offered by the database vendors should be considered.

The questions in this context are:

- What databases (DBs) are in use?
- Which DBs are in danger of running out of support?
- Which application system uses which DBs?
- What are the costs for operating and licensing of which DB?
- Which DBs are to be replaced and which application systems are affected thereby?
- Which organizational units host affected application systems?

The information needed for simulating this scenario consists of data about the infrastructure elements and their relationships of type *hosted at* to the organizational units, the technology - entities containing the database version and lifecycle information, and the relationships between application systems and infrastructure elements. The first deliverable to be created visualizes the use of DBs by the application systems. The same information should be exported as a tabular report (see deliverable in Figure 4.47).

App Id	App Name	DB ID	DB Name	Affected
100	Online Shop	100	MySQL Munich	x
200	Inventory Control System	200	Oracle Munich	
300	Monetary Transaction System (Germany)	200	Oracle Munich	
350	Monetary Transaction System (Great Britain)	500	DB2 London	x
400	Product Shipment System (Germany)	300	Oracle Hamburg	x
500	Accounting System	500	DB2 London	x
600	Costing System	200	Oracle Munich	
700	Human Resources System	200	Oracle Munich	
800	Data Warehouse	200	Oracle Munich	
900	Fleet Management System	200	Oracle Munich	
1000	Business Traveling System	300	Oracle Hamburg	x
1100	Document Management System	300	Oracle Hamburg	x
1200	Supplier Relationship Management System	300	Oracle Hamburg	x
1300	MIS (Management Information System)	200	Oracle Munich	
1400	Financial Planning System	100	MySQL Munich	x
1500	Campaign Management System	300	Oracle Hamburg	x
1600	POS System (Germany/Munich)	200	Oracle Munich	
1620	POS System (Germany/Hamburg)	300	Oracle Hamburg	x
1650	POS System (Great Britain)	500	DB2 London	x

Figure 4.47.: Scenario *Infrastructure Management*: Report showing application systems and databases affected by the consolidation

The database consolidation shall take into consideration the support periods offered by the database vendors. Especially, we focus on the databases running out of support before January 2012. This information shall be represented in a visualization (similar to the deliverable in Figure 4.48) as well as in a tabular report (see Figure 4.49).

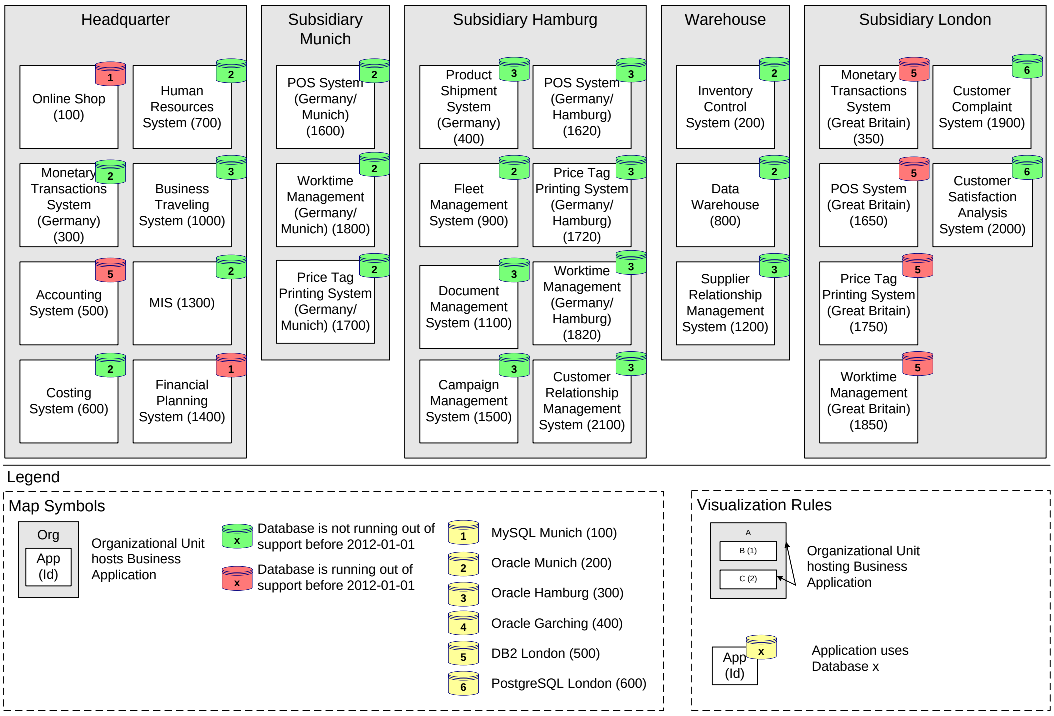


Figure 4.48.: Scenario *Infrastructure Management*: Cluster map visualizing databases running out of support and the application systems using them

id	name	start introduction	end introduction	start production	end production	start phase out	end phase out	end of support period
100	MySQL 2.0	01.01.2007	01.06.2007	01.06.2007	01.06.2009	01.06.2009	01.06.2010	31.12.2010
200	Oracle 9i	01.04.2005	01.04.2006	01.04.2006	01.01.2014	01.01.2014	31.12.2014	31.01.2015
500	DB2 6.0	01.06.2004	01.06.2005	01.06.2005	31.12.2009	31.12.2009	31.12.2010	30.06.2011
600	PostgreSQL 6.0	01.08.2007	01.03.2008	01.03.2008	01.01.2012	01.01.2012	31.12.2012	31.12.2012

Figure 4.49.: Scenario *Infrastructure Management*: Report showing databases running out of support

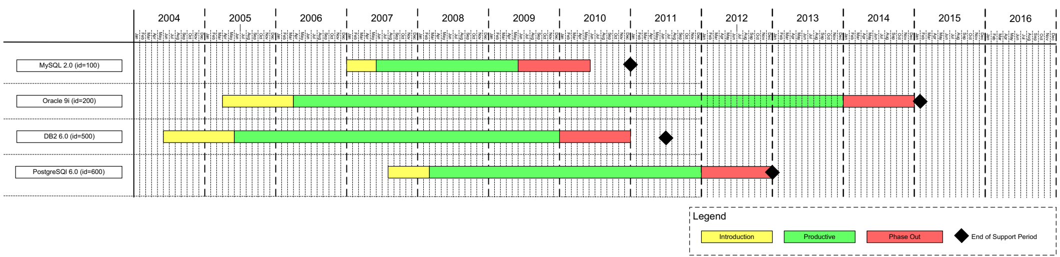


Figure 4.50.: Scenario *Infrastructure Management*: Interval map visualizing the lifecycle stages of the database technologies

In order to consider the planned support periods for the database systems more information about the lifecycle status and the expected future development of the databases is needed and shall be displayed in a diagram (see Figure 4.50 for an example). In a last step the databases to be removed during the consolidation process are selected (through an external process, e. g. a management decision). The repository should be updated with this information and the adapted information shall be displayed by annotating one of the visualizations above (as done in Figure 4.51) as well as by a tabular report.

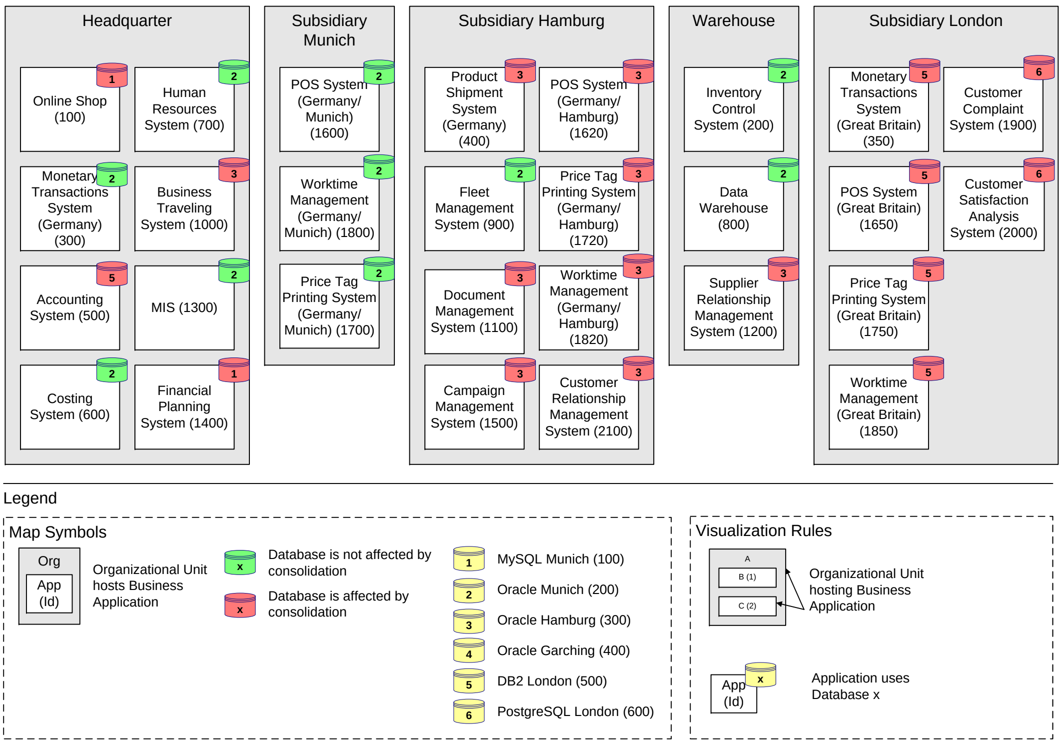


Figure 4.51.: Scenario *Infrastructure Management*: Cluster map visualizing application systems and databases affected by consolidation

4.2.10. Scenario Simulation

In order to ensure the consistency of evaluation for all tools the procedure for simulating scenarios is described in detail. For every scenario the following information is gathered:

Achievement of objectives: Was it possible to create the deliverables the scenario was aiming at (or comparable deliverables)?

Tool handling: How many pitfalls and shortcomings led to a high effort in producing the deliverables?

Procedure consistency: Does the procedure for creating the deliverables correspond to the methodology the tool is relying on or have parts of the tool's model be *misused* to simulate a scenario?

Procedure integration: Does the tool provide an integration of the activities and objectives in respect to other relevant activities and objectives of EA management and other simulated scenarios, etc.?

The results of scenario simulation are finally consolidated by the evaluation team in a workshop leading to an order of the tools reflected in the ordinal scaled values for the tools that are shown on the kiviat axes.

CHAPTER 5

Adaptive (Adaptive EAM)

Contents

Adaptive Inc has a long time background in the market for metadata repositories, therefore approaching the topic of EA management from that perspective. Therefore, the evaluated tool *Adaptive EAM 5.0* should according to Adaptive not be considered a standalone EA management tool, but as an integration tool, which can access data from and make accessible data to various other tools. Therefore, auxiliary tools e. g. the *discovery tool* are part of the Adaptive solution family. This tools allows to automatically discover a comprehensive set of properties of a computer system such as *number of CPUs*, *amount of memory*, or *running processes*.

Complementing the data and integration centric approach of *Adaptive EAM* strongly facilitated by the *Adaptive Integrator* component, the tool also provides several visualization functionalities strongly focused on read-only analyses. Not assuming, that the information visualized is also governed in *Adaptive EAM*, but is more likely to be aggregated from different sources, graphical modeling is not considered a major part of the Adaptive approach. Nevertheless, *Adaptive EAM* comes also shipped with an integration to Microsoft Visio utilizing stencils, by which graphical modeling can be performed.

5.1. Evaluation of Specific Functionality

Below the results of simulating the scenarios from Section 4.1 with the Adaptive EAM are detailed.

5.1.1. Importing, Editing, and Validating

The *Adaptive EAM* Tool provides among others, e. g. Microsoft Project and *ProVision*[1], native access to information stored in Microsoft Excel 97-2003 format. In importing the tool can fairly flexibly import information stored in different worksheets. This is especially caused by the fact, that the tool can perform a number of transformations during import. These transformations can be defined by an experienced user by using the annotation mechanisms of Microsoft Excel to provide XSLT code describing the transformation rules. Thereby, the full expressiveness of XSLT as a language for declaratively describing rules for processing of information is leveraged. Further, *Adaptive EAM* can access information stored in the XMI format and, via utilizing the *Adaptive Integrator*, from various other information sources including amongst others UML, BPM, and systems management tools. For synchronizing the information between the different tools, *Adaptive EAM* provides flexible mechanisms, that allow the user to define a synchronization policy on class level. In addition, unique identifiers leverage the seamless integration of different data sources. In importing the conformity of the data imported to the underlying information model is checked, in case of inconsistencies an error report is created.

For editing the information governed in the repository of *Adaptive EAM*, the tool provides input forms (see Figure 5.1), which automatically adapt to match the information model, i. e. provide additional input fields or dropdown boxes, if further attributes are introduced. The data within the repository can be accessed using the tree-like navigator, which provides an easy to comprehend overview on the information stored in the tool. If batch-editing of multiple objects in the repository is concerned, the user can employ e. g. Microsoft Excel or Microsoft Visio export mechanisms, perform the changes there, and re-import the data into the tool. Object consistency is therein ensured using unique object identifiers, while simultaneous changes of objects can be resolved using version merges as further detailled in Section 5.1.5. Furthermore, it is possible to employ the embedded Microsoft Excel Web Component to perform in-place editing concerning a set of objects (see Figure 5.2).

Finally, an additional mechanism for importing data into the *Adaptive EAM* repository has to be noted, the *discovery* mechanism. Using this, it is possible to automatically collect basic information

[1] In this respect, it has to be noted, that both repository information and various types of diagrams can be imported from *ProVision*.

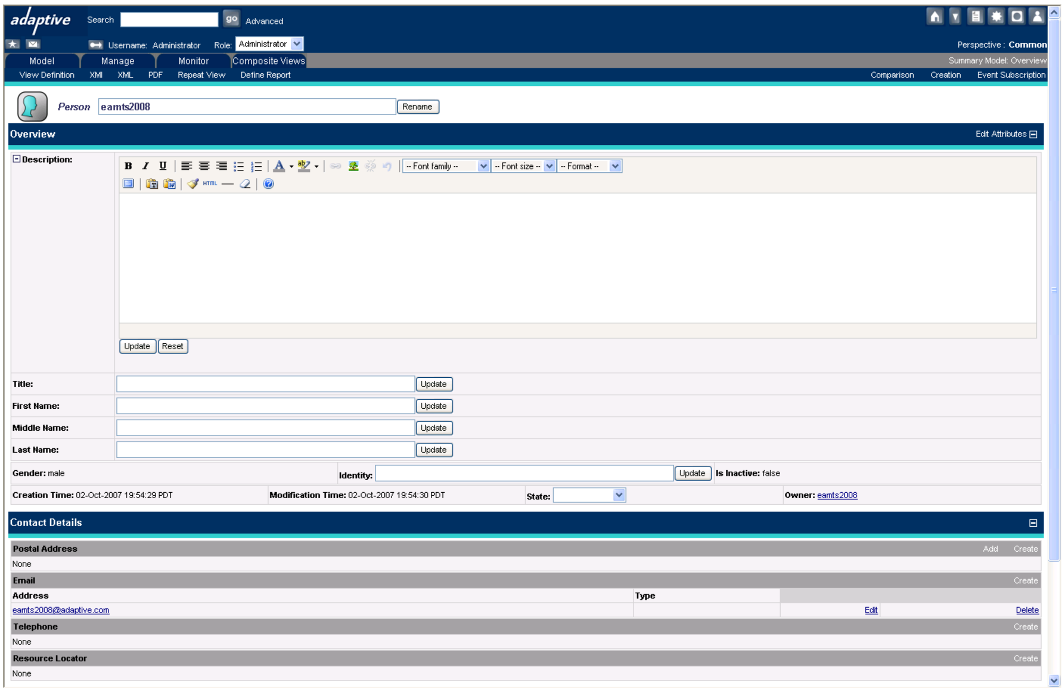


Figure 5.1.: *Adaptive EAM* - axis Importing, Editing, and Validating: Edit dialog for an object

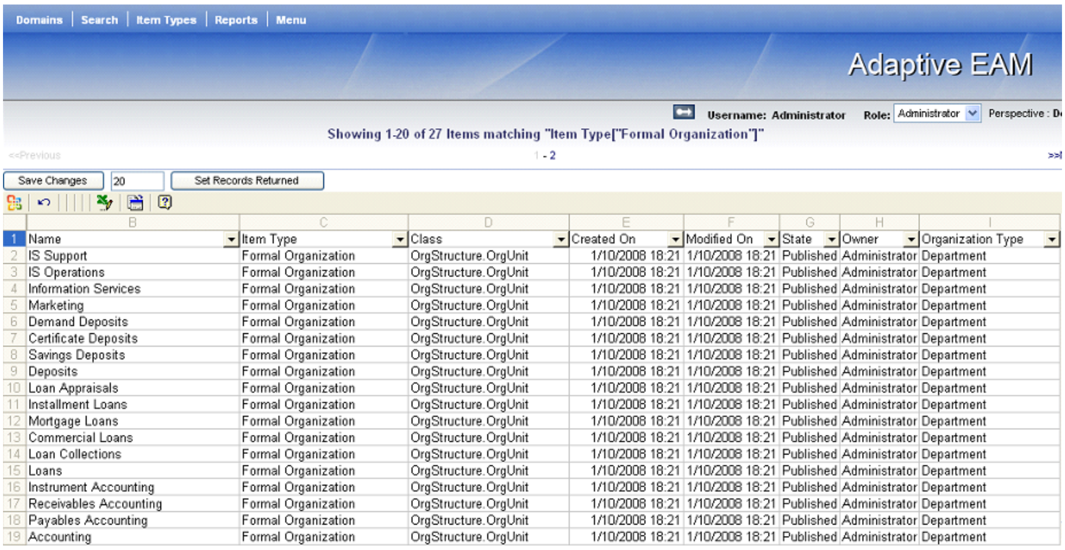


Figure 5.2.: *Adaptive EAM* - axis Importing, Editing, and Validating: In-place edit of objects using the Microsoft Excel Web Component

on computer system, e. g. concerning the CPU, RAM, or installed programs. The information gathered can subsequently be imported into the repository of *Adaptive EAM* or can be graphically visualized in the discovery tool. As the *discovery* tool is shipped as an add-on to the core *Adaptive EAM*, we subsequently do not detail on the functionality provided by this tool.

Rating: 6

5.1.2. Creating Visualizations

Concerning the creation of visualizations *Adaptive EAM* provides different layouting options, although the diagrams are mainly *Graphlayout Maps* according to our terminology (cf. Section C). These visualizations are created in real-time from the underlying repository data, filtered on classes and relationships by user demand. As a layout option the user can choose to visualize relationships as lines (cf Figure 5.3) or via nesting of symbols leading to a cluster map visualization as the one shown in Figure 5.4. Nevertheless, *Process Support* and *Time Interval Maps* cannot be created automatically by *Adaptive EAM* as out-of-the-box, while the user can create such visualizations from the underlying repository data manually, employing the Microsoft Visio stencils as supplied with the tool.

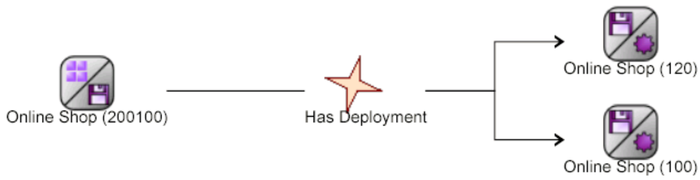


Figure 5.3.: *Adaptive EAM* - axis Creating Visualizations: Automatically generated graphlayout map

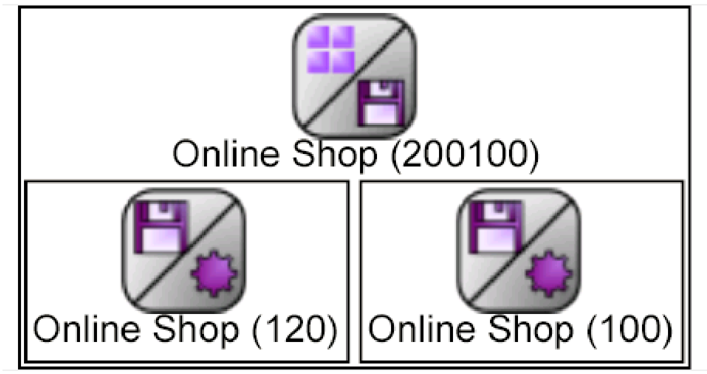


Figure 5.4.: *Adaptive EAM* - axis Creating Visualizations: Automatically generated cluster map

Portfolio matrices and *Swimlane diagrams* as demanded in the scenario cannot automatically be created, although for the matrices the exporting functionalities to Microsoft Excel can be used to indirectly achieve the creation. Additionally, the Microsoft Visio templates allow to create visualizations of that kind, as does the integration to *ProVision*, from which diagrams created employing the *Business Swimlane modeler* can be imported into the repository of *Adaptive EAM* and therefore included in the visualizations.

Rating: 4

5.1.3. Interacting with, Editing of, and Annotating Visualizations

Concerning the interaction with visualizations, the two different kinds of creating visualizations with *Adaptive EAM* have to be considered separately. At first we focus on the automated visualization creation within the web-based user interface, later referring to the Microsoft Visio templates.

The visualizations as created in the web interface of *Adaptive EAM* are mostly automatically layouted SVG diagrams representing the current state of the information repository and are thereby not intended to be edited. Thus, they do not provide direct support for editing user interaction. Nevertheless, the user can zoom and pan visualizations directly in the browser window, leveraging the vector graphic zooming capabilities as provided by the SVG viewer. Furthermore, the symbols representing objects from the repository are linked to the pages actually displaying the corresponding object's information. Additionally, different mechanisms to further refine the set of objects displayed can be applied - different layouting algorithms, such as *spring layout* and *tree layout* may be used to change the visual make up. Furthermore, user can decide on the basic direction, the diagram should conform to, e. g. *left-to-right* or *top-to-bottom*. Concerning the depiction of object relationships, the user can choose whether to display a symbol representing the relationship or not; also the labels of the lines corresponding to the relationships can be shown or hidden. Finally, specific visualizations in the web-interface support further adaptations especially concerning the visibility of objects and attributes via the so called *Graph Browser Interface*, see Figure 5.5.

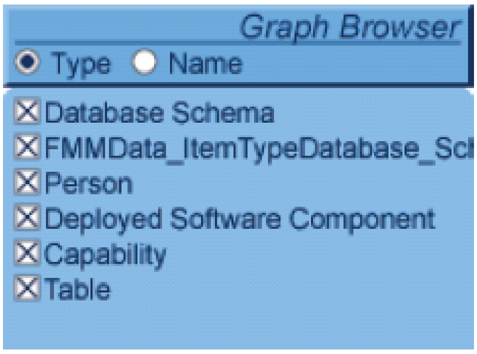


Figure 5.5.: *Adaptive EAM* - axis Interacting with, Editing of, and Annotating Visualizations: Graph browser interface

Manual adaptations, e. g. concerning the color coding of the symbols are not directly possible in the automatically created visualization in the web-interface. Nevertheless, a user or consultant having experience with programming XSLT transformations could leverage such functionalities by defining an appropriate transformation file for creating color-coded visualizations. These transformations are part of the customization of *Adaptive EAM*, which is offered by *Adaptive*.

As an alternative to the SVG based visualization, *Adaptive EAM* also allows to create graphlayout diagrams (see Figure 5.6). These visualizations also provide strong support for non-editing interactions, as e. g. moving symbols or navigating by clicking on the symbol representing a certain concept. Furthermore, changes in the perspective used, can be easily applied. Nevertheless, these visualization do not support performing semantic changes to the information contained in the underlying repository, leaving these changes to be done in the textual description of the corresponding objects.

The visualizations created using the Microsoft Visio templates provide the full-scale flexibility of Microsoft Visio for changing the visual make up. Therefore, the user can freely adapt the visualization by moving, resizing, or changing the colors of the symbols contained. Nevertheless, these changes do not always exert semantic changes in the underlying repository, as e. g. no predefined positioning related semantics is supported. The visualizations manually adapted by using the Microsoft Visio templates can subsequently be presented in the web-interface.

Rating: 2

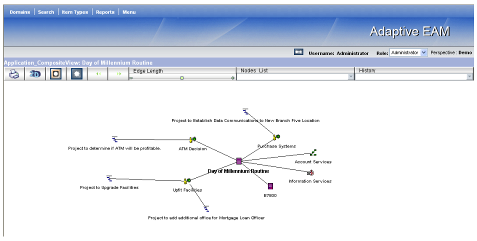


Figure 5.6.: *Adaptive EAM* - axis Interacting with, Editing of, and Annotating Visualizations: Interactive graphlayout visualization

5.1.4. Flexibility of the Information Model

Adaptive EAM comes shipped with a set of compulsory standard information models (for an overview see Figure 5.7), e. g. for UML, CWM, or the Enterprise Distributed Object Computing, which are all intended for specific use cases. Additionally, these models are linked together, where appropriate and form the overall information model of adaptive, which counts a few hundred of classes and thousands of relationships. Nevertheless, this information model can be adapted freely by the user, leveraging the full-scale modeling capabilities as provided by the Meta Object Facility (MOF) of the OMG. Therefore, the user can introduce new classes, attributes, and relationships as well as association classes or enumeration types. Additionally, cardinality constraints and default values can be applied.

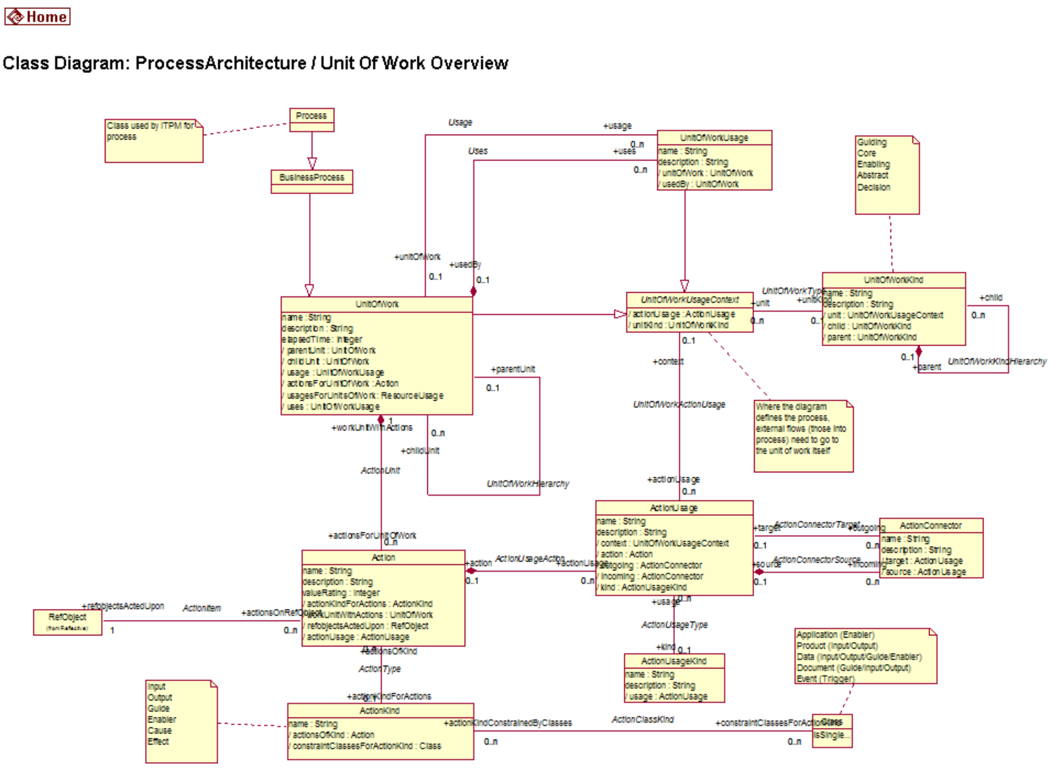


Figure 5.7.: *Adaptive EAM* - axis Flexibility of the Information Model: Excerpt from the predefined information model of *Adaptive EAM*

The information modeling capabilities of *Adaptive EAM* are further complemented by the possibility to import any MOF or ECore compatible metamodel into the tool and its repository. Subsequently, the user can access and leverage all the editing and visualization techniques as alluded to in Sections 5.1.1 and 5.1.2 based on the newly introduce information model. Nevertheless, such full-scale adaptations (or replacements) of the tool's information model had according to adaptive not to be performed often, as the predefined information model provides most concepts needed for EA management out-of-the-box. For minor adaptations, especially concerning attributes, the user can introduce so called *classification schemas* (see Figure 5.8), defining a set of connected attributes, which can be assigned to any class from the information model, which is thereby extended. This mechanism is further leveraged by the *Intelligent Classification Editor*, providing the capability to apply a certain classification programmatically and rule based (see Figure 5.9) to a large number of objects from the repository. Therefore, a newly introduced classification schema can be populated with data easily, enabling quick reaction to changing information demands, as long as this information is deductible by rules. Finally, the current information model of the tool or parts thereof can be exported to files in UML-XMI format as well as re-imported, and can thus be used and visualized with the common UML tools.

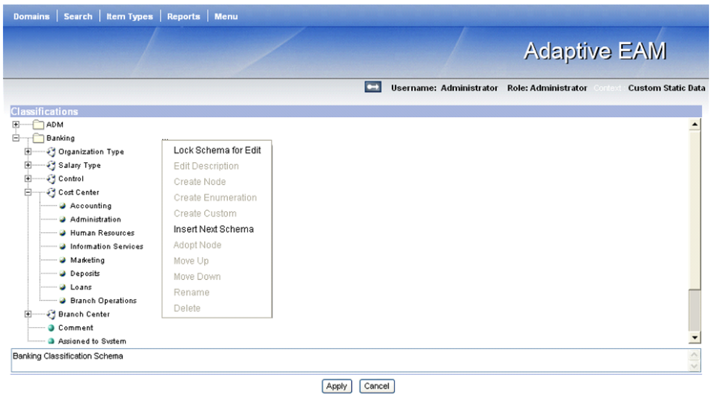


Figure 5.8.: *Adaptive EAM* - axis Flexibility of the Information Model: Editing a classification schema

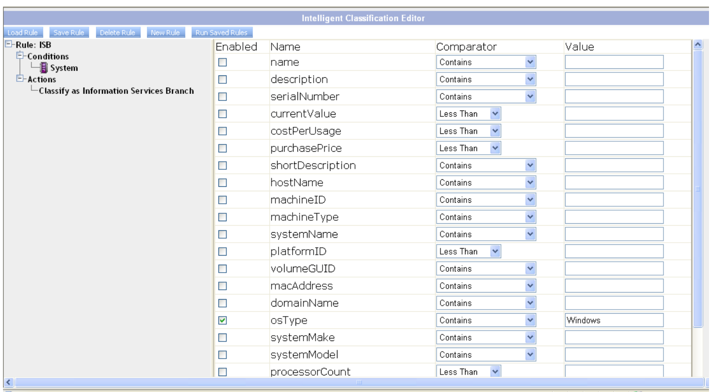


Figure 5.9.: *Adaptive EAM* - axis Flexibility of the Information Model: Rule editor for the *Intelligent Classification Editor*

Rating: 7

5.1.5. Communication and Collaboration Support

Adaptive EAM is a full thin client solution, of which the functionalities are accessible via the web interface. The underlying back-end application is strongly designed for scalability and can be distributed on different web servers for load balancing, therefore applicable for a large and distributed user community. In order to facilitate the collaborative creation and manipulation of information about the EA, the system provides an integrated workflow system, which can track a state for every object within the repository. According to the workflow states different actions are required to take place, e. g. the approval of an edited object by a designated user. Further mechanisms of that kind could be created by customization.

Concerning the historization of changes applied to the data governed within the tool, *Adaptive EAM* provides sophistacated versioning mechanisms, enabling the user to trace, which changes have been performed by whom. Eventing mechanisms, which fire, when a model or an element thereof is adapted, can be used to trigger automated validation of the data as well as for notification of users. Furthermore, if an object has been changed simultaneously by different users, *Adaptive EAM* can support the merge of the different versions, thus providing a powerful conflict resolution mechanism. This approach is taken instead of providing explicit locking mechanisms on application level. Locks have only to be established (cf. Figure 5.8), if parts of the information model as e. g. the classification schemas as alluded to in Section 5.1.4 are to be edited.

Adaptive EAM provides role-based access control to the different objects in the repository as well as to the different other elements in the navigation as e. g. stored queries (cf. Section 5.1.7). According to the defined access restrictions on the actual role, the user can access objects in read-only or in editing mode.

Additionally, *Adaptive EAM* provides a functionality also useful for collaborative scenarios. When classification schemas (see Section 5.1.4) have newly been introduced and should be populated with data, it possible to perform this task via e-mail. Thereby, information necessary for filling the corresponding attributes is distributed.

Rating: 6

5.1.6. Support of large scale Data

Adaptive EAM is capable of importing large scale data considerably fast, although no real-time performance can be achieved. Using the import wizard, the user can select the perspective, the information should be imported to. Once the information has been imported into the tool, no obvious reduction in interaction performance is experienced. Interacting with the different navigation options and the dialogs for editing information in the repository is not visibly affected.

For helping the user to deal with large scale data governed by the tool, some mechanisms are present. It is e. g. possible to apply sorting, which is especially helpful concerning lists containing a large number of objects. In addition, the Microsoft Excel Web Component allows the user to chose the number of objects to be displaying, which might be very useful for finding a specific object, especially if queries of low selectiveness are applied.

Finally, the visualization mechanisms, as far as supported in the tool (cf. Section 5.1.2) are not visibly affected by the number of objects in the repository. The same is true concerning the mechanisms for interacting with the information as far as the web frontend is concerned.

Rating: 5

5.1.7. Impact Analysis and Reporting

Adaptive EAM provides different mechanisms for determining objects, which have a specific property, from within the repository contents. The user can make use of a search form, which is commonly present in the different pages. The search is performed on the repository contents using full-text capabilities. Additionally, a recommender system exists, providing typing support.

A second and more sophisticated mechanism for querying the repository is the *Adaptive EAM* query builder. This dialog (see Figure 5.10) enables the user to select a base type, from which the querying should start. On the instances of this type the user can establish selections according to the values of the corresponding attributes. Furthermore, the user can navigate relationships to related objects and apply selections on their attributes. Also transitively navigating relationships of related objects is possible. While not all different types of aggregate functions are supported, the user can apply counting on the related objects.

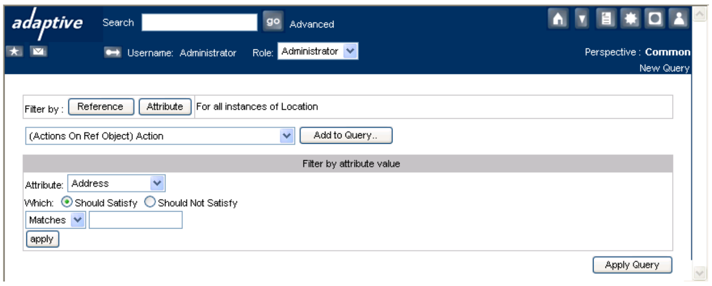


Figure 5.10.: *Adaptive EAM* - axis Impact Analysis and Reporting: Building a user defined query

The querying mechanism can be considered very powerful, although performing sophisticated ad-hoc queries requires a skilled user to create the actual query definition. In order to facilitate querying by occasional users without strong experience with the tool, *Adaptive EAM* provides the possibility to store a defined query and assign it to a given user role. Users of that role can then execute the query without having to edit the query definition. Nevertheless, parametrizing the stored queries is not directly supported by *Adaptive EAM*.

Another also very powerful mechanism for determining objects bearing specific properties is to utilize the embedded Microsoft Excel Web Component with its grouping and filtering functionalities. Therein, the user can choose attributes to e. g. classify application systems by the date of their start of production. Additionally, the information can easily be exported to *fat-client* Microsoft Excel, further leveraging a compulsory set of functionalities, e. g. pivot-tables.

Impact analyses can also be performed utilizing the *Adaptive Relational Access Module (RAM)*. Via this module, it is possible to access information stored in the repository of *Adaptive EAM* in a relational format, i. e. as views in a database management system. This technique allows to leverage the full extent of functionalities as provided by the *Standard Query Language (SQL)* to define queries against the relational views on the information. While although external SQL-enabled reporting tools could be used to execute these queries, it possible to do the same in the *Adaptive EAM* platform, too. The results are then displayed in a report based on the Microsoft Excel Web Component (see Figure 5.11).

Finally, a predefined set of analyses is available via the *Dashboard* functionality leading to visualizations similar to the one shown in Figure 5.12.

Rating: 7

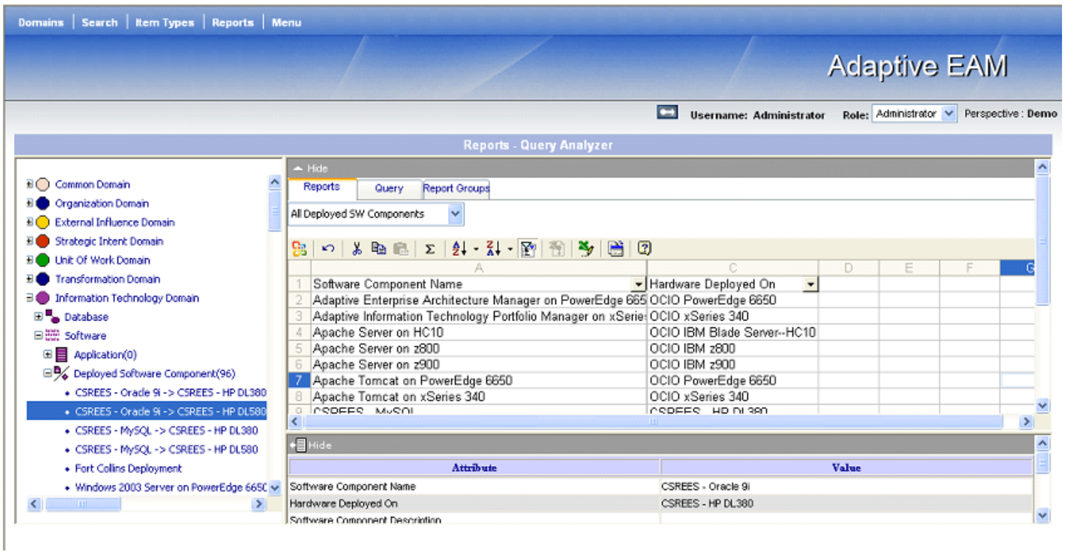


Figure 5.11.: *Adaptive EAM* - axis Impact Analysis and Reporting: Displaying the result of a RA Module query in the Microsoft Excel Web Component

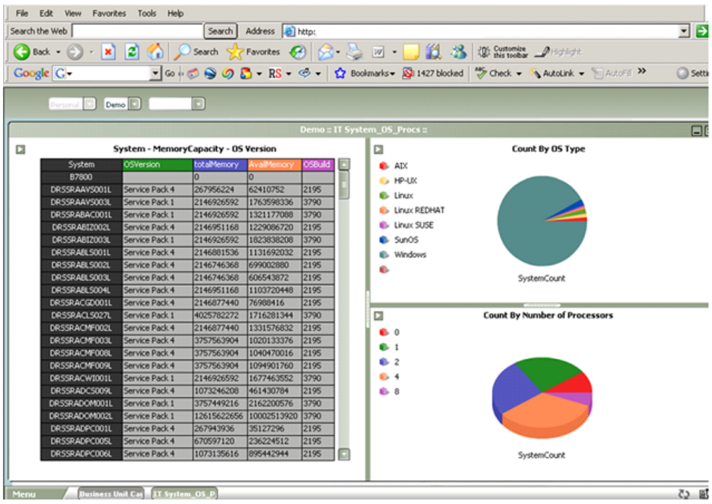


Figure 5.12.: *Adaptive EAM* - axis Impact Analysis and Reporting: A predefined *Dashboard* report

5.1.8. Usability

The user interface of *Adaptive EAM* can be considered very intuitive and well-structured, although a user logged on in an unrestricted profile might be overwhelmed by the sheer richness of options and dialogs offered. Problems connected to the user interface can be reduced for ordinary users by providing them with a profile that hides features not needed for accomplishing the EA management related tasks. Therefore, they are spared from navigation and editing possibilities, which they not need.

In addition to the intuitive user interface, the user can leverage functionalities common to many browser based applications, such as navigation via the back-button and the history function of the browser. Furthermore, bookmarking of pages is supported by the application, as the URLs of the pages are directly connected to a certain view on the information from the repository. Therefore, permanent linking of e. g. the result pages of searches or queries is possible, inspite of the fact that the underlying data changes, such that a user can re-query the repository with the same query again just by clicking on a bookmarked link. A similar functionality is used in the tool itself to supply a set of predefined clickable navigation maps 5.13 for structuring the concepts EA. Via customization the user can introduce his own navigation maps based on the same mechanisms.

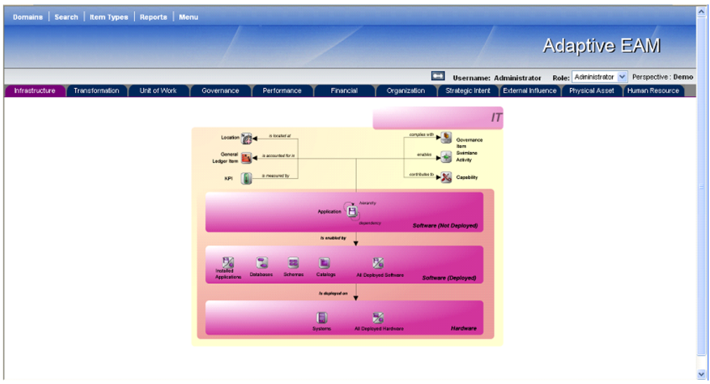


Figure 5.13.: *Adaptive EAM* - axis Usability: Clickable navigation map

The richness of the information model shipped with *Adaptive EAM* might sometimes lead to problems with finding the appropriate concepts for representing a specific element from the EA. Although a documentation of the predefined information model can be very helpful here, a detailled description of the concepts would be further useful. Nevertheless, consulting and training may also help the user to build up the necessary knowledge. The same is true concerning the importing mechanisms, which provide a lot of flexibility, while on the other hand not giving support by wizards for configuration. Preconfigured import files, containing examples on the possible operations, are here provided and can be considered very helpful for understanding the basic mechanisms, while training and support would be necessary to unleash all importing capabilities as provided by *Adaptive EAM*.

Rating: 4

5.2. Evaluation of EA Management Support

This section describes the results of the scenario simulation for EA management support.

5.2.1. Landscape Management

The *Adaptive EAM* infomation model defines a rich set of classes to represent all necessary objects making up an application landscape. For example, the information model contains classes to model organizational units, application systems, and business processes together with the neccessary references to express that an application system supports a process and a process belongs to an organizational unit. Nevertheless, the ternary relationship between these concepts, as realized by the *SupportRelationship* class from the *SoCaStore* information model (see Section 3.2), is not part of the predefined information model. Such a relationship could be introduced, making use of the functionalities for adapting the information model as presented in Section 5.1.4, and could subsequently be reported on, e. g. via the RAM functionalities.

In order to visualize, how an application landscape looks like, the visualization facility of *Adaptive EAM* can be leveraged to generate certain diagrams. Based on a selected process, all organizational units, which are related to this process and all application systems, which provide support for this process, can be shown on a diagram that is automatically generated. Figure 5.14 gives an example of such a diagram, starting with a process together with the organzational units the process belongs to and the application systems, which support this process.

Since the visualization is always based on a selected, single object (e. g. the business process in the example above), a visualization showing the entire application landscape is not directly supported. Gaining a holistic overview of the landscape is therefore possible by iteratively visualizing each process.

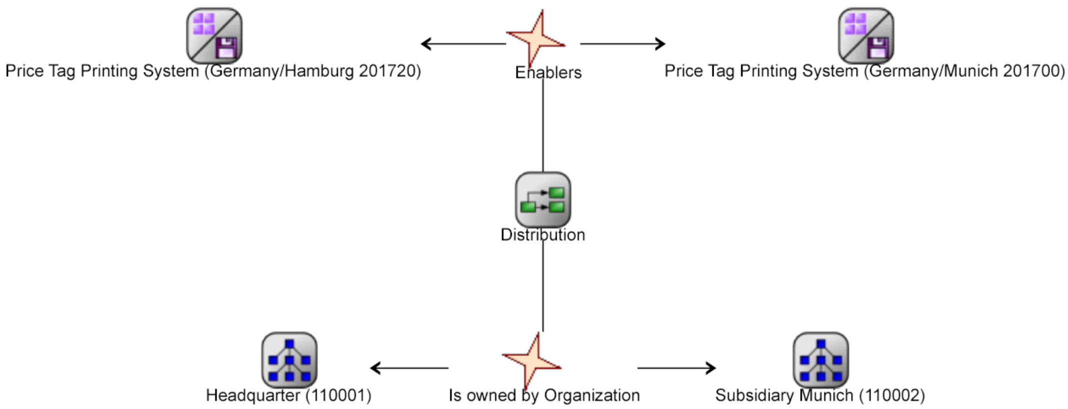


Figure 5.14.: *Adaptive EAM* - axis Landscape Management: Automatically generated map showing a process with associated organizational units and supporting application systems

Adaptive EAM introduces a mechanism called *perspectives* to maintain different versions for individual objects. A user can therein select the perpective, which he is currently working in. All information shown to the user is then related to his currently selected perspective and all changes he makes, only apply to this perspective. In addition, there are mechanisms for propagating this information to some other perspectives as well, which will be introduced below. All perspectives are created based on an exsisting perspective, called the root perspective. According to the *Adaptive EAM* terminology, the perspective, which is created, is called a *version* of its root perspective. A newly created version of a perspective inherits all objects, which exist in its root perspective. In contrast, newly created objects are at first only present in the currently selected perspective. Changes applied to an object in the root perspective are automatically propagated to derived versions of this perspective, if the corresponding object has not been changed so far. In contrast, changes to an object in a derived version are not propagated back to the root perspective.

In the context of this scenario, the perspective mechanism can be used to model a planned landscape based on the current landscape. In this case, the planned landscape should be a version perspective of the root perspective, which represents the current landscape. The perspective mechanism thereby protects the current landscape from changes applied in the context of the planned landscape. If subsequently the planned landscape becomes the current one and a new landscape should be planned, a version perspective of the formerly planned landscape can be created. Creating perspectives in this way establishes a chain of dependencies, which preserves all former *current* landscapes. This dependency chain can be visualized by *Adaptive EAM* like shown in Figure 5.15. Temporal information can be assigned to individual systems in the application landscape or to landscape versions in general, which could subsequently be used in dashboard reports specifically designed for this kind of evaluation.

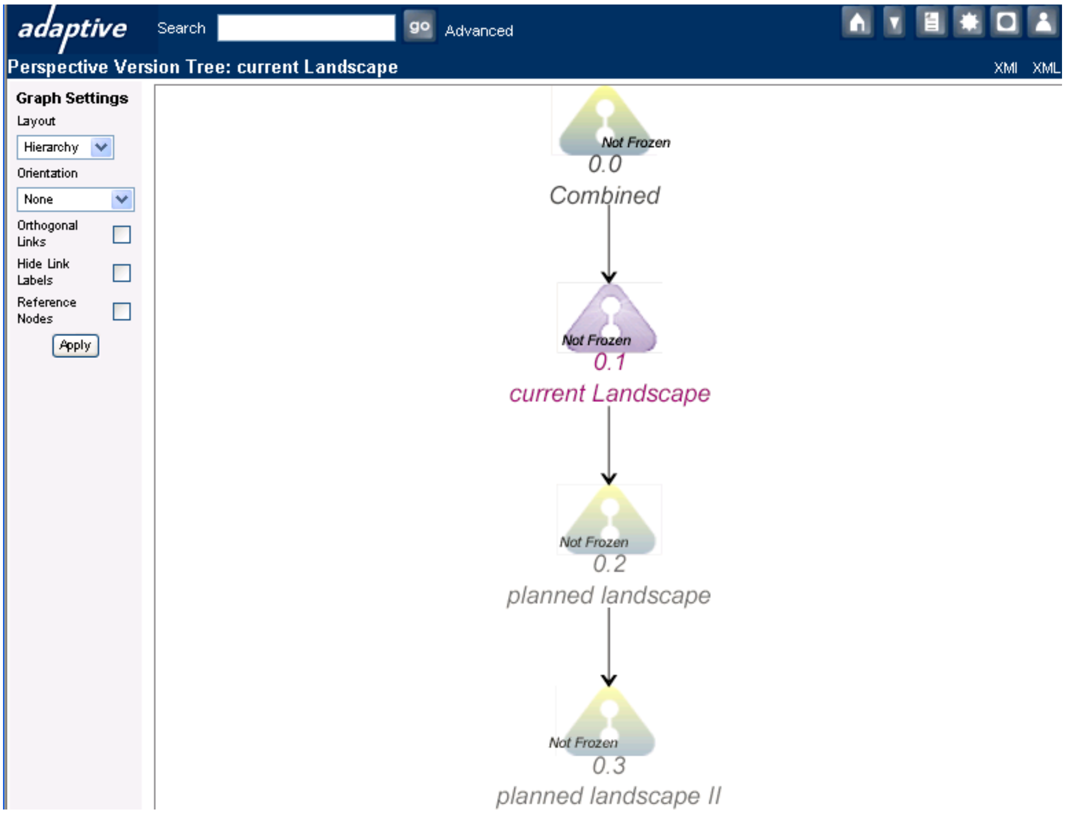


Figure 5.15.: *Adaptive EAM* - axis Landscape Management: Dialog showing the dependencies between different perspectives

The perspective mechanism provides an object comparison facility to generate an overview on the changes applied to a selected object between two different perspectives. The comparison facility considers all attributes of the object as well as all relationships to other objects. The comparison can regard one object, as well as all objects of a certain type, e. g. all application systems. The comparison of one object results in a dialog that contains only the changed attributes, where the changes are highlighted by a color-coding depending on which value is assigned in which perspective. An example for a comparison result is shown in Figure 5.16.

The powerful mechanisms, as alluded to above, are further complemented by the possibility to assign projects to perspectives, thereby modeling a project as being the driver of a transition between different perspectives (landscapes). This mechanism can further be used to derive information in the context of project portfolio management. Additionally, the different perspectives of the landscape can be compared based on certain metrics via the dashboard functionality as described in Section 5.1.7.

Procedure consistency is mostly given as landscape planning can be performed by creating perspectives. Nevertheless, a *SupportRelationship* for considering the evolution of the process support is not supported out-of-the-box.

Procedure integration is mostly given. Information entered in previous scenarios can be reused, although a transition from landscape versions to projects is not directly supported, only vice versa.

Rating: 4

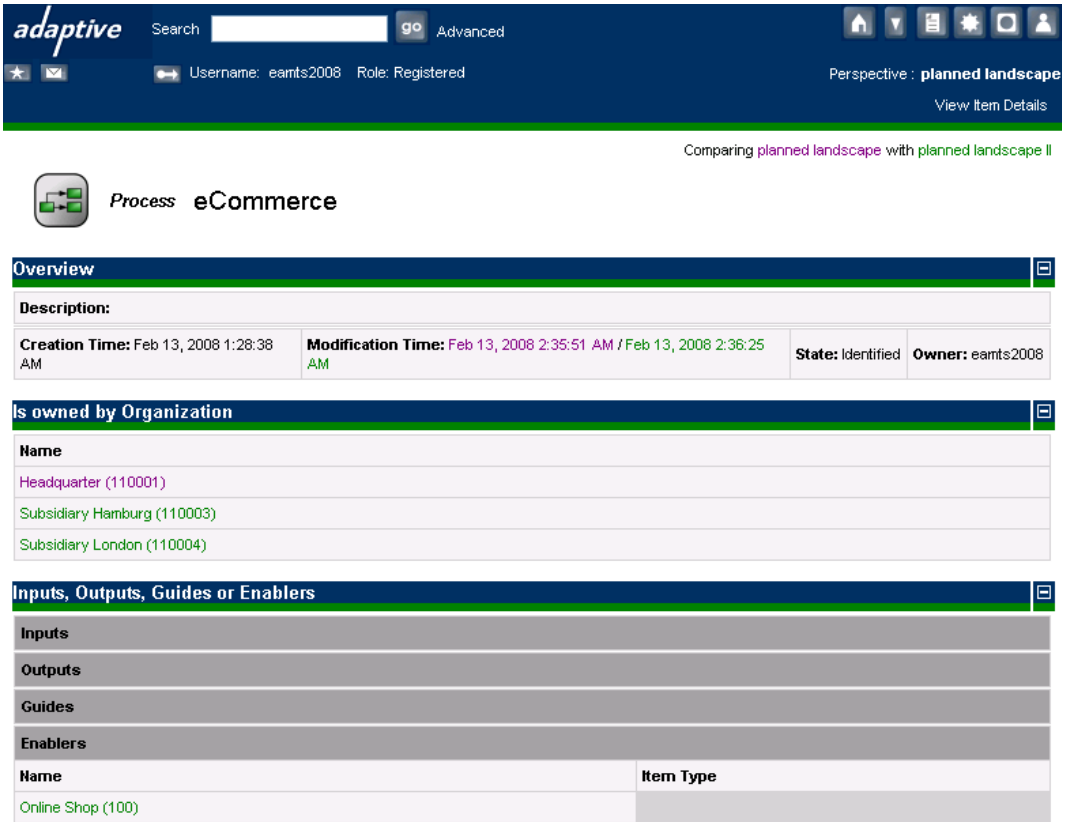


Figure 5.16.: *Adaptive EAM* - axis Landscape Management: Comparison dialog showing changed attributes and changed relationships of a selected object

5.2.2. Demand Management

Although no direct suppot for demand management is provided by *Adaptive EAM*, the tool offers comprehensive functionalities to integrate project managment tasks backed by concepts from the repository. The information model provides the *project* class to model projects. As a consequence, demands can be modeled as *project tasks*, since the *project task* class is related to the *project* class and offers a relationship in order to model, which application systems are affected by this task. Futher, a user defined classification can be used to indicate, whether a specific *project task* represents a demand, preventing the user from misinterpreting the concepts.

To generate an overview, which demands have been received in a selected period, the advanced search facility is advisable to use. As *Adaptive EAM* stores the creation and modification time of all objects in the repository, an advanced search restricted to the time period of interest and to the class, which is used to model demands, results in a list of demands received in the respective period. Figure 5.17 shows an example of these settings.

For expressing that certain demands lead to a project, the *part of project* relationship can be used, although here the additional usage of a user defined classification schema is advisable, in order to prevent misinterpretations. In this way, the deduction of projects based on the received demands is carried out by creating a new project directly from the detail page of the currently designated demand or by creating a new reference to a project, which already exists so far. Nevertheless, this mechanism introduces a modeling restriction, setting a demand to be associated with just one specific project. The deduction of projects can be visualized by utilizing the diagram generation facility, which results

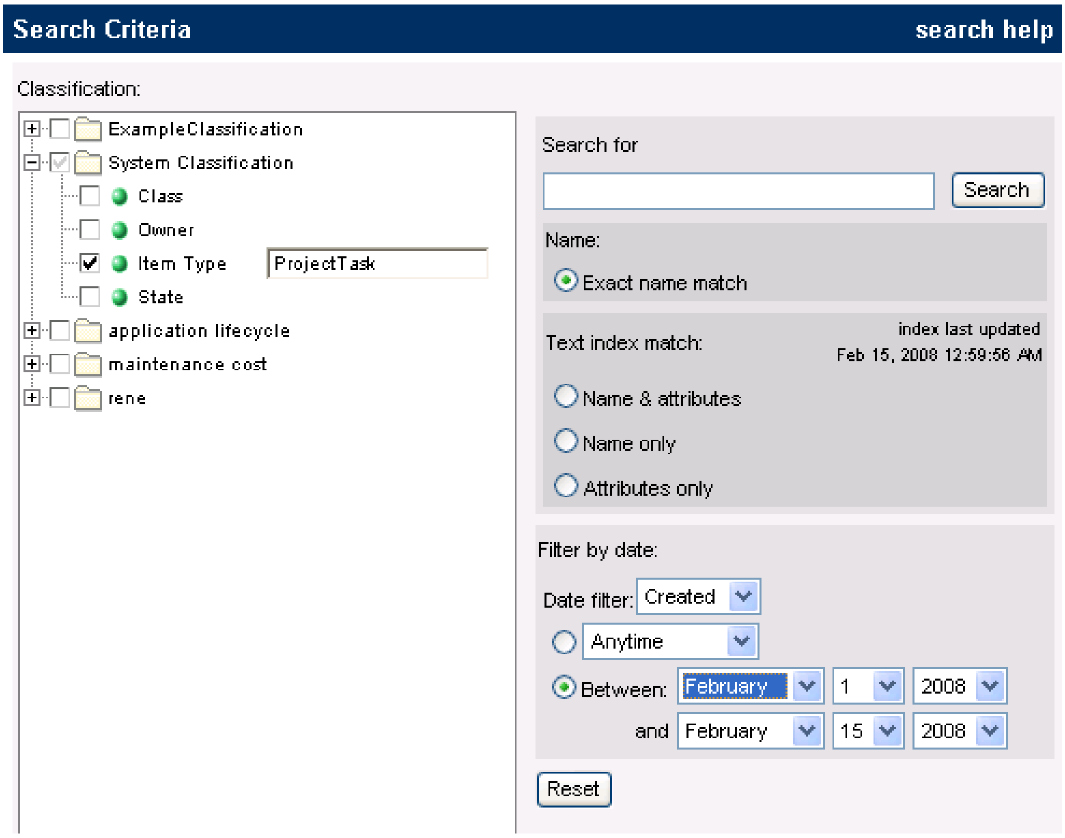


Figure 5.17.: *Adaptive EAM* - axis Demand Management: Advanced search settings to list all received demands

in a visualization that shows the assigned project of the demand. Figure 5.18 depicts an example of such a visualization.

Furthermore, the user can leverage the impact analysis mechanism for evaluating the project, a demand has lead to. This information is available on the detail page of the demand and a user can easily navigate to the project by using the provided link.

Additionally, the class *project task*, which is used to model demands, supports a relationship called *received item*. This relationship can be used to model, which application system is affected by a demand. Again, the diagram generation facility can be used to generate a graphical overview that shows, which application systems are affected by a demand. Further, the impact analysis, based on the detail page of a demand, provides another way to evaluate, which application systems are affected by the selected demand and allows to access these application systems by traversing a link. Figure 5.19 depicts an example of a diagram that shows, which application systems are affected by a demand.

Contrasting the approach taken above, two other ways to map demands to the information model could be taken. Firstly, demands could be represented as projects supplied a specific classification. Using this mapping, the user would additionally have to maintain project tasks detailing the demand in order to create the relationship between a demand and the affected application systems. The second way employs an adaptation to the information model going beyond the application of classification schemas. Herein, the user would have to introduce a new class for demands, both related to projects and application systems.

Procedure consistency is mostly given. As no concept directly complementing the demand is given in *Adaptive EAM*, the user has to pay attention in appropriately using the mapped classes.

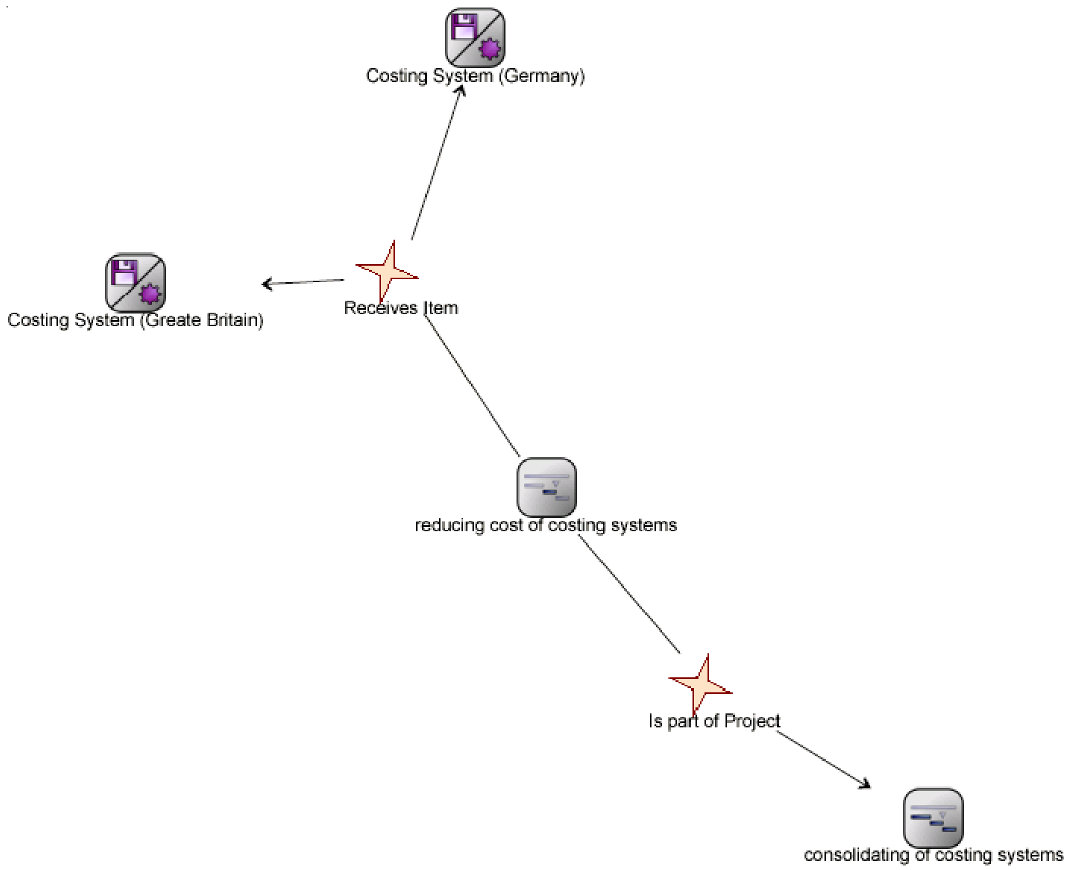


Figure 5.18.: *Adaptive EAM* - axis Demand Management: Visualization of a demand, which is assigend to a project

Procedure integration is mostly given. Information maintained in previous scenarios can be reused in the context of this scenario, although demands cannot be seamlessly transformed to projects.

Rating: 4

5.2.3. Project Portfolio Management

In the context of this scenario, similar considerations as in Section 5.2.2 have to be taken, as *Adaptive EAM* actually supports projects, but does not allow them to be directly linked to application systems. Therefore, the more detailed project description utilizing objects of type *project task* have to be employed. We have chosen to use this concept instead of using actual projects for reasons of simplicity – nevertheless, the user could in accordance to the methodology of *Adaptive EAM* decide to maintain both project and project task information.

For this scenario, the projects are modeled as instances of the class *project task*, of which instances have already been used to represent demands. As a consequence, the classification schema, as introduced in Section 5.2.2, is again facilitated to foster the distinction between these distinct concepts.

To get an overview about, which projects have been received in a period of interest, the advanced search facility of *Adaptive EAM* can be used. An advanced search can be restricted to the classification of project proposals and to the time peroid of interest regarding object creation, in order to retrieve all project proposals, which have been created in this period.

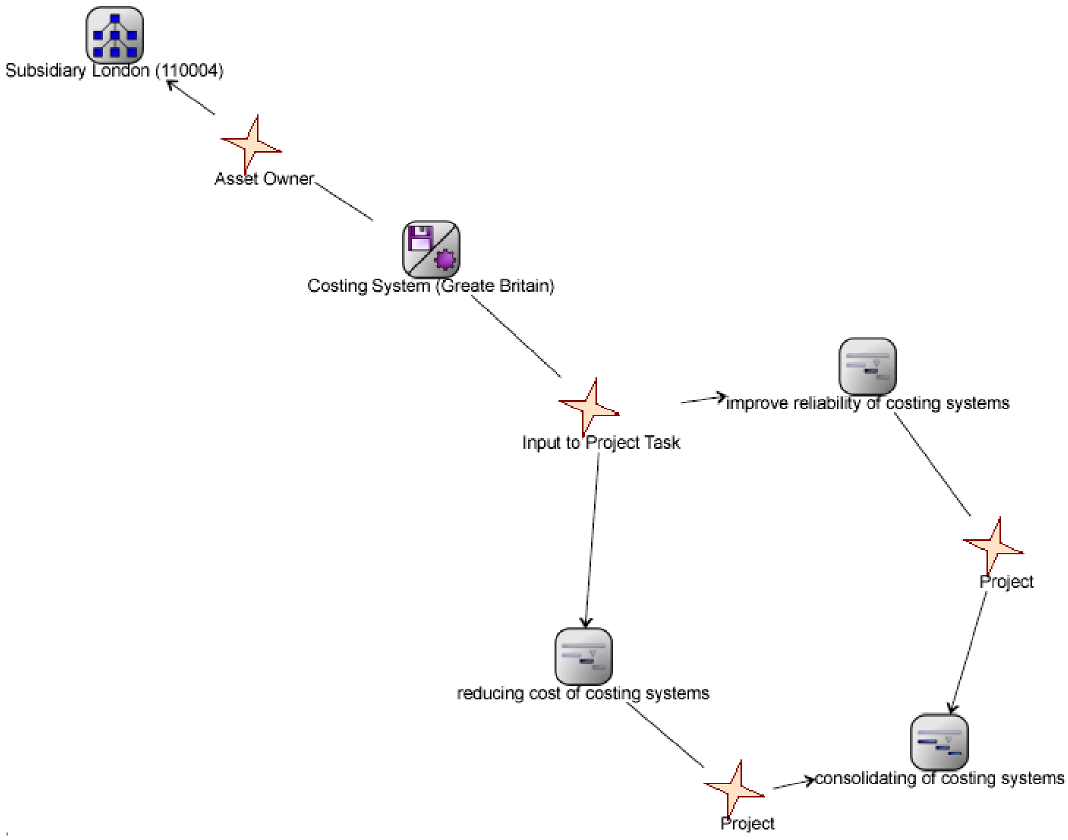


Figure 5.19.: *Adaptive EAM* - axis Demand Management: Visualization of a demand showing to which project the demand is assigend and which application systems are affected by the demand

Facilating the concept of classification schemas, the user could annotate, which project proposals have to be accomplished in any case, by providing an appropriate boolean attribute. Figure 5.20 shows an example of this classification. The advanced search mechanism can subsequeently be used to evaluate against this classification, in order to retrieve the projects, which must be accomplished in any case. Further, this evaluation can be combined with the one described above, to find out all obligatory projects recieved in a certain time period.

When considerations, which application systems are affected by a selected project should be facilitated, an impact analysis starting at the specific project can be performed. Alternative to this more manual approach, a diagram showing a project and all of its related application systems can be created. If, as alluded to above, the option to model both projects and project tasks in this context is chosen, a similar diagram showing a project with its related project tasks and all applications systems, which are affected by these tasks can be created automatically. Figure 5.21 gives an example of a visualisation of a project together with its related tasks and the affected application systems.

Deriving possible conflicts in the project portfolio, due to the same application system being affected by more than one project at the same time, in a tabular report can be performed employing the reporting functionalities of the RAM (cf. Section 5.1.7). Furthermore, analyses of that kind could also be executed via the Microsoft Excel Web Component and could further be visualized leveraging the direct export to *fat-client* Microsoft Excel (see Figure 5.22), facilitating native visualization techniques as bar charts available there. As a final option to derive possible project conflicts, the user could

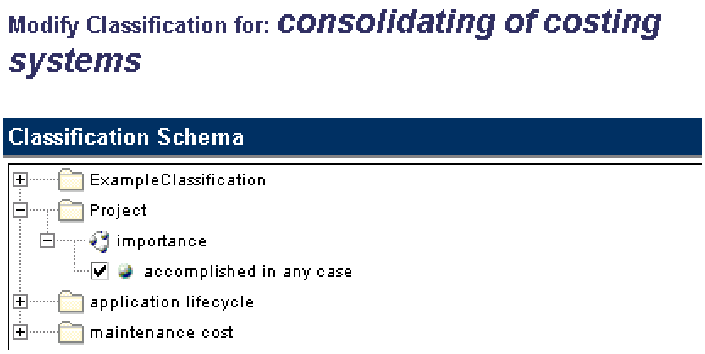


Figure 5.20.: *Adaptive EAM* - axis Project Portfolio Management: Dialog for classifying the importance of a project

create an ad-hoc graphlayout map (cf. Figure 5.6 in Section 5.1.3 for an example) centering around an application system suspected to be a potential source of conflicts.

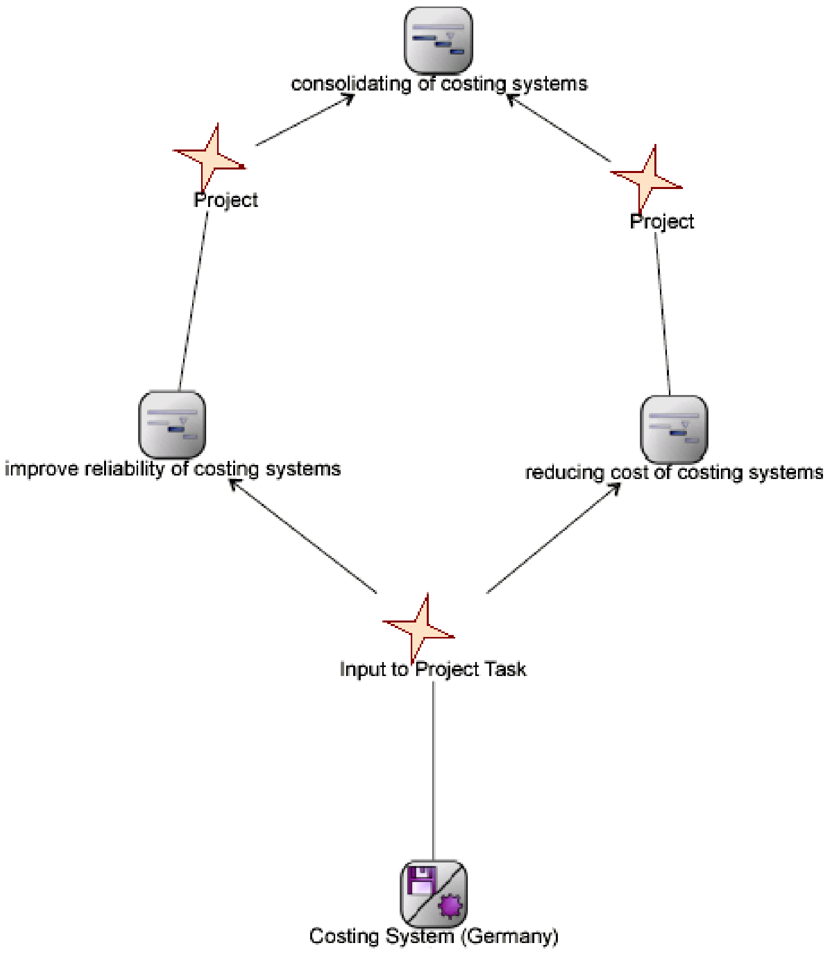
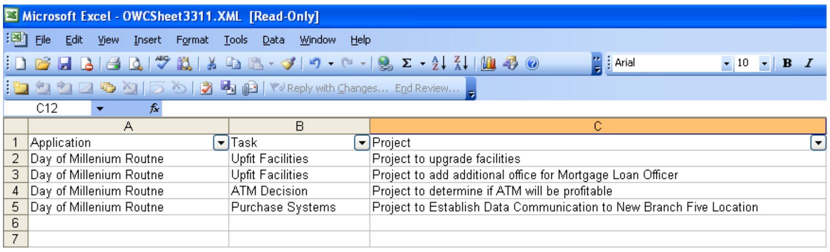


Figure 5.21.: *Adaptive EAM* - axis Project Portfolio Management: Two project tasks related to one project



Application	Task	Project
Day of Millenium Routne	Upfit Facilities	Project to upgrade facilities
Day of Millenium Routne	Upfit Facilities	Project to add additional office for Mortgage Loan Officer
Day of Millenium Routne	ATM Decision	Project to determine if ATM will be profitable
Day of Millenium Routne	Purchase Systems	Project to Establish Data Communication to New Branch Five Location

Figure 5.22.: *Adaptive EAM* - axis Project Portfolio Management: Tabular report in the Microsoft Excel indicating potential project conflicts

In addition to the derivation, as alluded to above, the information model of *Adaptive EAM* provides a class (*Project Task Interdependency*) for explicating known dependencies between two or more project tasks. Thereby, it is possible to assign relationships to interdependent project tasks from a given interdependency instance, further supplying information on the type of interdependency in the attribute values of the corresponding instance.

Procedure consistency is mostly given. Projects can be modeled utilizing two modeling options, with one of them demanding the user to detail projects down to the project task level.

Procedure integration is mostly given. Seamless integration between demand and project portfolio management is not directly given, due to the mapping alternatives for demands and project, as alluded to above.

Rating: 4

5.2.4. Synchronization Management

Project dependencies resulting from affecting the same application system or the same organizational unit can be derived manually by performing impact analyses. These analyses can be executed, starting form the projects, which should be examined, traversing their relationships to the assigned project tasks and subseqeuently to the affected application systems. More automation concerning this deduction of dependencies could be achieved utilizing the RAM for by applying an appropriate query against the thereby provided relational repository view, taking into account both affected application systems and temporal aspects of projects. The result of this report could subsequently be displayed utilizing the Microsoft Excel Web Component.

Nevertheless, explicated interdependencies between project tasks can be modeled using instances of *Project Task Interdependency* or by utilizing the relationships *is required by* or *depends on project*, which are provided by the *project* class. These two relationships can be used to model complex dependencies between certain projects. Furthermore, *Adaptive EAM* supplies a visualization assistant facilitating the visualization of these relationships. Figure 5.23 shows an example thereof.

Temporal aspects of projects, e. g. planned start or end dates are not directly part of the predefined information model in *Adaptive EAM*, but could nevertheless be supplied relying on a user defined classification schema. The same is true, if project delays should be considered, such that the information from the classification could be used to determine conflicting project schedules, if the project interdependencies had been maintained in the relationships, as introduced above.

Procedure consistency is mostly given. Project conflicts and interdependencies can be deduced via a user defined query in the RAM.

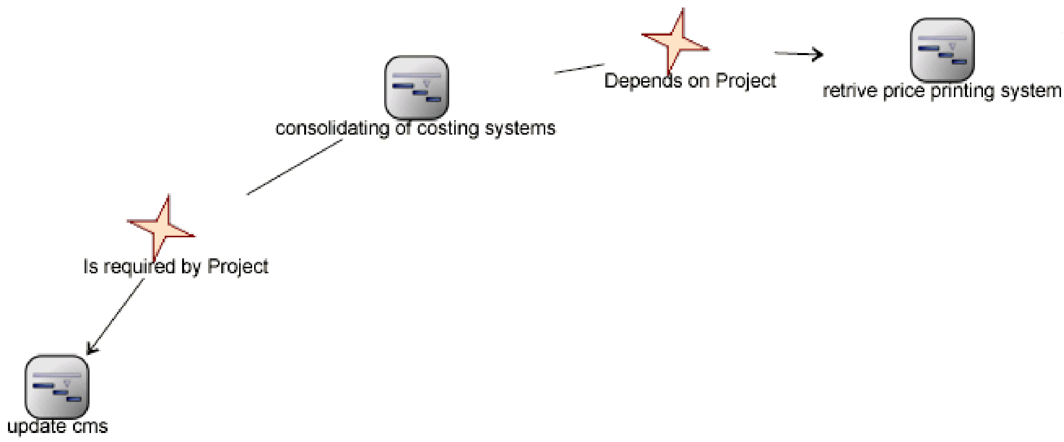


Figure 5.23.: *Adaptive EAM* - axis Project Portfolio Management: Visualization of complex project dependencies

Procedure integration is mostly given. Information previously entered into the tool can be reused in this scenario, although temporal project dependencies have to be explicated.

Rating: 4

5.2.5. Strategies and Goals Management

Adaptive EAM provides strong support for strategy and goals management via the predefined information model, which supports both strategies and goals (via the *objective* class). Furthermore, objectives in *Adaptive EAM* can be subsequently composed into sub-objectives contributing to the operationalization of the corresponding goal. Using the built-in visualization facility, it is possible to visualize strategies and objectives in a diagram showing their relationships. Figure 5.24 gives an example of such a diagram showing a strategy together with its related objectives.

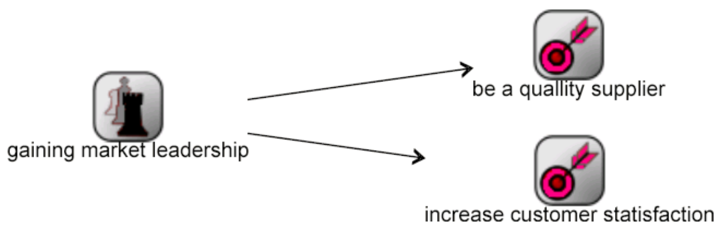


Figure 5.24.: *Adaptive EAM* - axis Strategies and Goals Management: Diagram of a strategy together with its related objectives

In addition to that, the information model provides the possibility to maintain the relationship between a project and an objective, thus further pursuing the operationalization. Using these relationships, an impact analysis can be performed for determining which application systems are a affected transitively by which goal. The same information could also be displayed graphically, using an ad-hoc graphlayout visualization similar to the one shown in Figure 5.25.

For supplying information on the level of fulfillment concerning a goal (an objective), it is advisable to introduce a user defined classification schema providing an enumeration attribute representing the fulfillment. Thereby, the user can evaluate all goals concerning their fulfillment by creating an advanced search restricted to the required fulfillment value (cf. Figure 5.26).

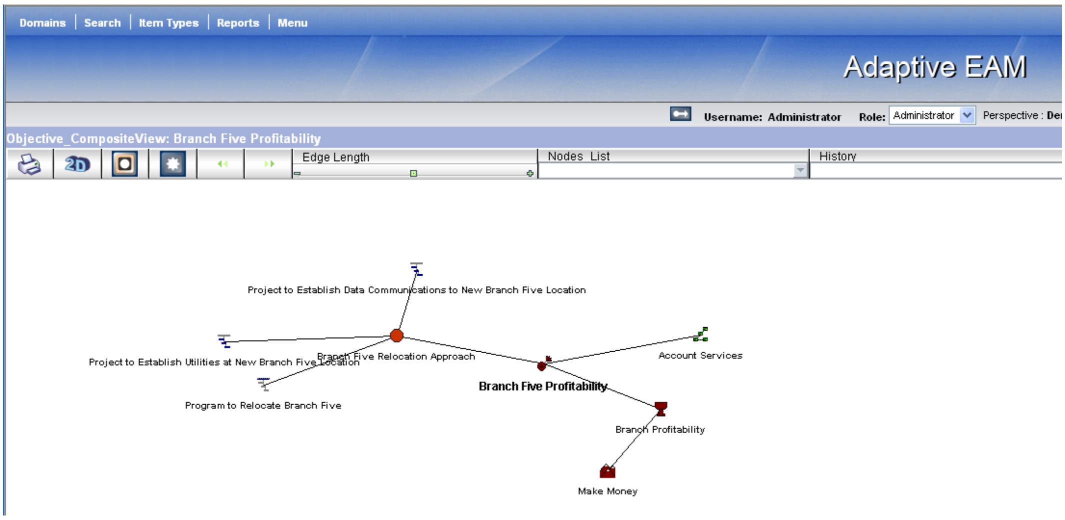


Figure 5.25.: *Adaptive EAM* - axis Strategies and Goals Management: Graphlayout map displaying the relationships of strategies, goals, projects, and application systems

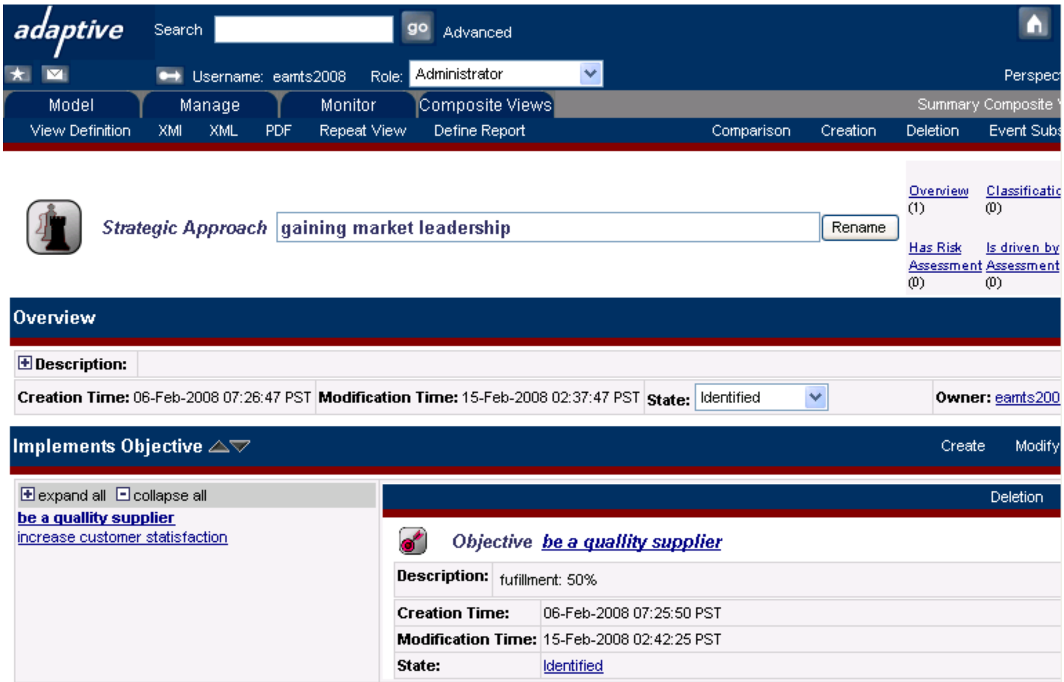


Figure 5.26.: *Adaptive EAM* - axis Strategies and Goals Management: Detail page of a strategy showing the fulfillment of the related objectives

Procedure consistency is completely given. The user can traverse from strategies *downwards* to application systems or the other way round.

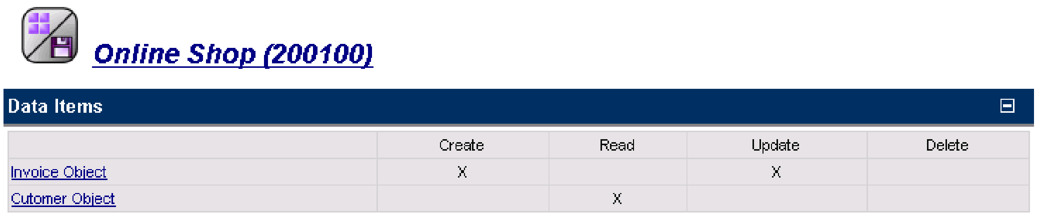
Procedure integration is completely given. Information previously entered into the tool can be reused.

Rating: 5

5.2.6. Business Object Management

The information model of *Adaptive EAM* provides support for business object management based on the class *Data Type* that is suitable for representing business objects. The *Data Type* class can be referenced by application systems, representing that an application system makes use of the corresponding business object. Further, the user can supply one or more modes of interaction concerning an application system and the business object. Available usage modes are here *create*, *read*, *modify*, and *delete*.

Using the detail page of an application system, the user can access a summary of all related business objects together with the usage modes the systems exerts on the corresponding objects. A tabular overview on the business objects used by an application system together with the mode of usage can further be generated from this page, Figure 5.27 shows an example of this overview.


Online Shop (200100)

Data Items

	Create	Read	Update	Delete
Invoice Object	X		X	
Cutomer Object		X		

Figure 5.27.: *Adaptive EAM* - axis Business Object Management: Tabular overview about used business objects and their access modes

Considerations in reverse order, e. g. concerning the application systems affecting instances of a certain business object, the user can facilitate the detail page of the corresponding *Data Type*. Therein, information on the usage modes is not directly accessible. Figure 5.28 shows an example of a diagram visualizing by which application systems a business object is accessed.

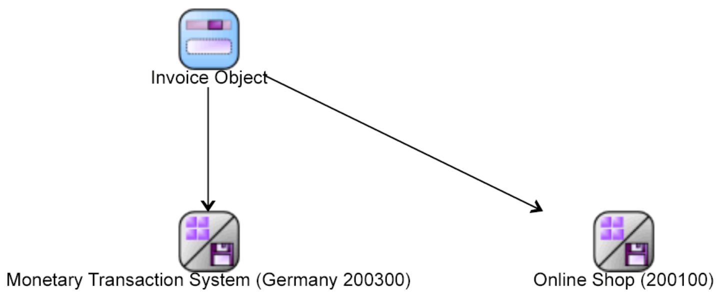


Figure 5.28.: *Adaptive EAM* - axis Business Object Management: Visualization of the application systems, which use a business object

Procedure consistency is mostly given, as information on business objects can be maintained and mostly visualized, although information on actual usage modes of business objects cannot directly be visualized.

Procedure integration is completely given, as information previously entered into the tool can be reused.

Rating: 6

5.2.7. SOA Transformation

The SOA transformation scenario is supported by *Adaptive EAM* by a combination of certain facilities. The custom classification facility offers the possibility to maintain information on the number of different domains employing the specific applciation systems. This information could also be maintained utilizing a relationship to a newly introduced domain class and its instances, an approach not taken during scenario simulation. Therefore, a self defined classification schema can be configured in *Adaptive EAM* to define the respective attributes. These attributes are available for every application systems in the detail page. An advanced search can be facilitated to evaluate these classification attributes. See Figure 5.29 for an example of an appropriate advanced search configuration. The corresponding classification could be further augmented to also supply an attribute for maintenance costs.

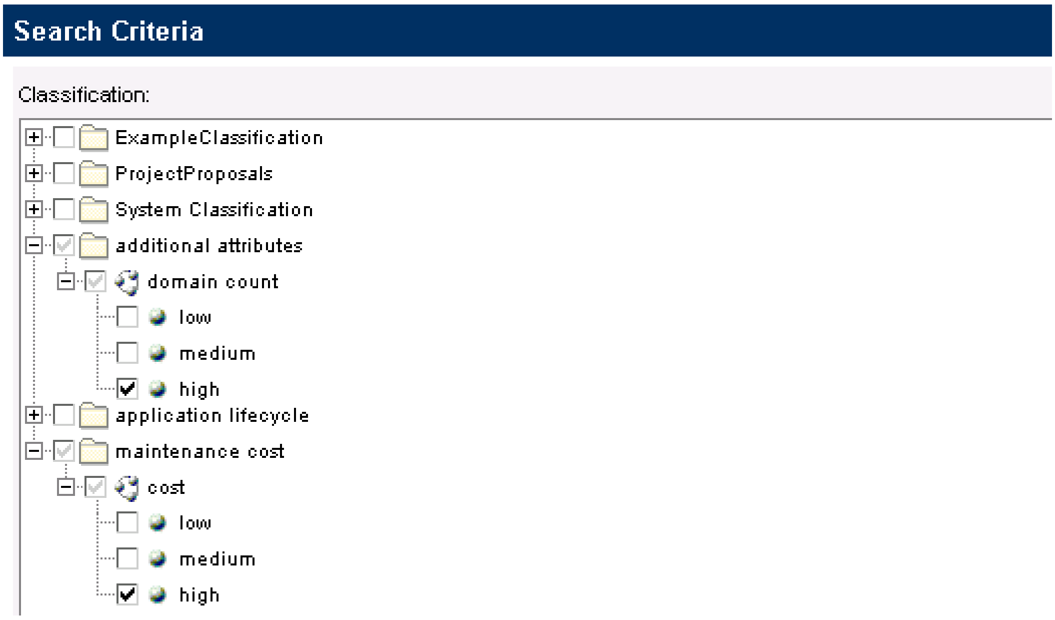


Figure 5.29.: *Adaptive EAM* - axis SOA Transformation: Example of an advanced search configuration to retrieve application systems, which are used at a large number of domains and exert high maintenance cost

The automatically generated visualizations are not directly adaptable to support color-coding or textual annotations for supplying information on the number of domains. Nevertheless, via reporting similar information can be drawn from the underlying repository.

The application systems, which have been selected to be transformed to services, can be assigned to projects, which are intended to actually perform these transformations. If a classification schema is applied to the corresponding project, the user can easily trace, which service results from which application system via a SOA transformation project. An advanced search can then be used to deliver a list of all projects which are related to the SOA transformation. Furthermore, the usage of projects to model the transformation of an application system into a service allows to explicate dependencies between different transformations. Using the project dependency relationships (see Section 5.2.4), it is possible to express that a transformation must be completed before another transformation can start.

Procedure consistency is mostly given. The needed information can be supplied using classification schemas. Nevertheless, visualizations for facilitating reflections on service candidates are not directly available.

Procedure integration is completely given. Information previously entered into the tool can be reused.

Rating: 5

5.2.8. IT Architecture Management

Adaptive EAM supports IT architecture management via classes representing *architectural blueprints* and *solutions*, with the first class mapped to *Standards* and the latter one to *Procedures*[2] from the information model of the tool. Both classes inherit from the class *Compliance Item*, which supplies a relationship *complies with compliance item* to the class application system. This relationship can be used to model that an application system is realized in accordance to an architectural blueprint or solution. For supplying the build-up of a solution, the *consists of compliance items* relationship can be utilized for decomposing the *standard* to smaller elements corresponding to the *SolutionElements* from the *SoCaStore* information model (cf. Section 3.2). While doing this, the introduction of a classification schema for distinguishing solutions and their constituents is advisable to prevent misinterpretations.

Having performed a mapping of concepts, as alluded to above, the detail page of the application system can be used to access information, which architectural blueprint this application system implements. Further, an impact analysis can be performed to evaluate, which architectural solutions the implemented architectural blueprint is based on. A more automated way to evalutate at which architectural solution and solution elements an application system is based on, is to generate a visualization of these relationships using the visualization assistant. An examplary visualization is shown in Figure 5.30.

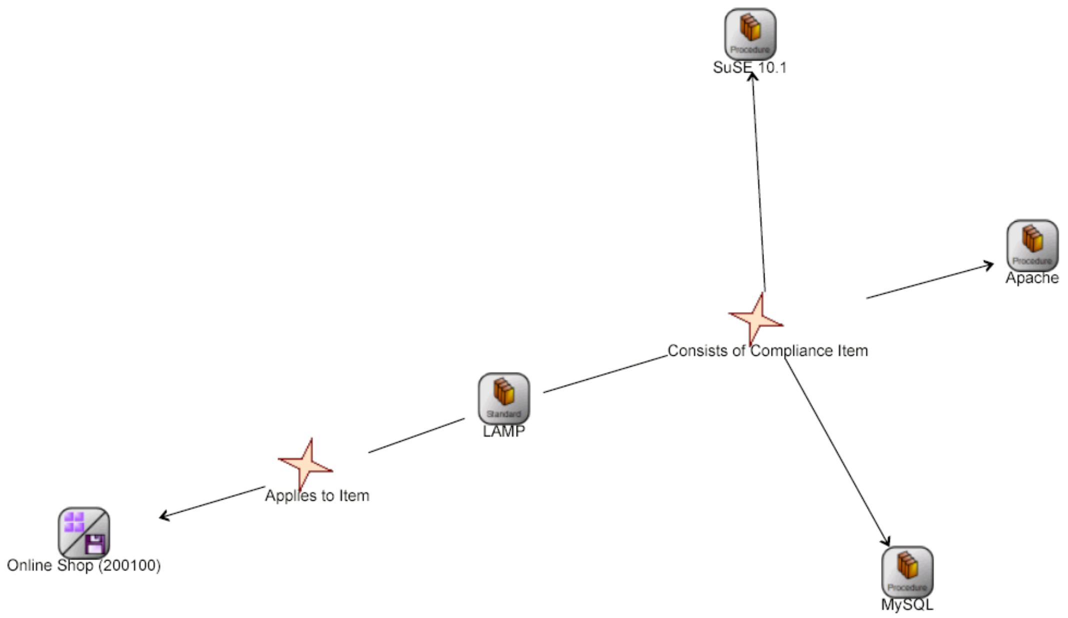


Figure 5.30.: *Adaptive EAM* - axis IT Architecture Management: Example visualization of an architectural solution together with its related application systems and the solution elements

If consideration on the application system based on a specific architectural blueprint should be made, the list of corresponding application systems can be accessed in the detail page of the architectural

[2] It would also be possible to map both concepts to the class *Standard*. This approach would leverage decomposition as demanded in this scenario, but also lead to further potential for misinterpretation.

blueprint. Also corresponding solutions can be displayed in the detail page. The visualization assistant of an architectural blueprint object supports the generation of a diagram showing the application systems, which are implemented using the corresponding architectural blueprint.

Considerations on the retirement of a solution can be performed utilizing a custom classification schema. Figure 5.31 shows an example of an architectural solution, which is annotated to be replaced in the corresponding classification schema attribute. By using the advanced search mechanism of *Adaptive EAM*, all architectural solutions, which are selected as to be retired, can be listed. An iterative impact analysis over all solutions, which should be retired, delivers the list of application systems affected by this retirement.

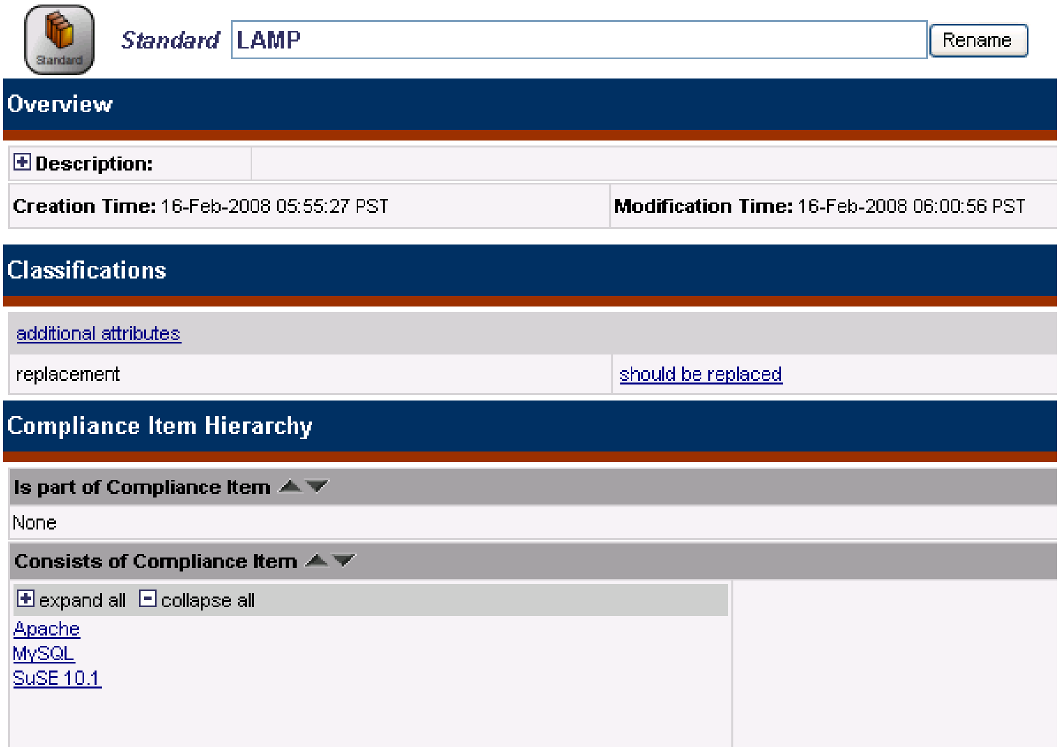


Figure 5.31.: *Adaptive EAM* - axis IT Architecture Management: Example of an architectural solution, which classifies as to be retired

Finally, a custom query can be used to identify architectural solutions, which are not used by an application system. For this purpose, the custom query mechanism of *Adaptive EAM* offers the posibility to evaluate the quantity of a relationship, in this context the number of associated application systems.

Procedure consistency is mostly given. While architectural solutions can be supported, their constituents have to be mapped similarly, leading to possible misinterpretations.

Procedure integration is completely given. Information previously entered into the tool can be reused.

Rating: 6

5.2.9. Infrastructure Management

The central concept of infrastructure management, the *InfrastructureComponent*, can be mapped to the *technical application* as supported in the information model of *Adaptive EAM*. The application systems are associated to instances of *technical application* via the relationship *depends on software component*. This information can be maintained in the detail page of the respective application system, where these dependencies are also shown. Furthermore, a visualization of these dependencies can be created. If transitive dependencies are concerned, *Adaptive EAM* provides convenient support, as a technical application can be modeled to be dependent on another technical application, e. g. a database management system can be modeled to be dependent on a specific operating system. These transitive dependencies are also shown in the detail page of an application system (see Figure 5.32 for an example) as well as at the graphical visualization of them (see Figure 5.33).

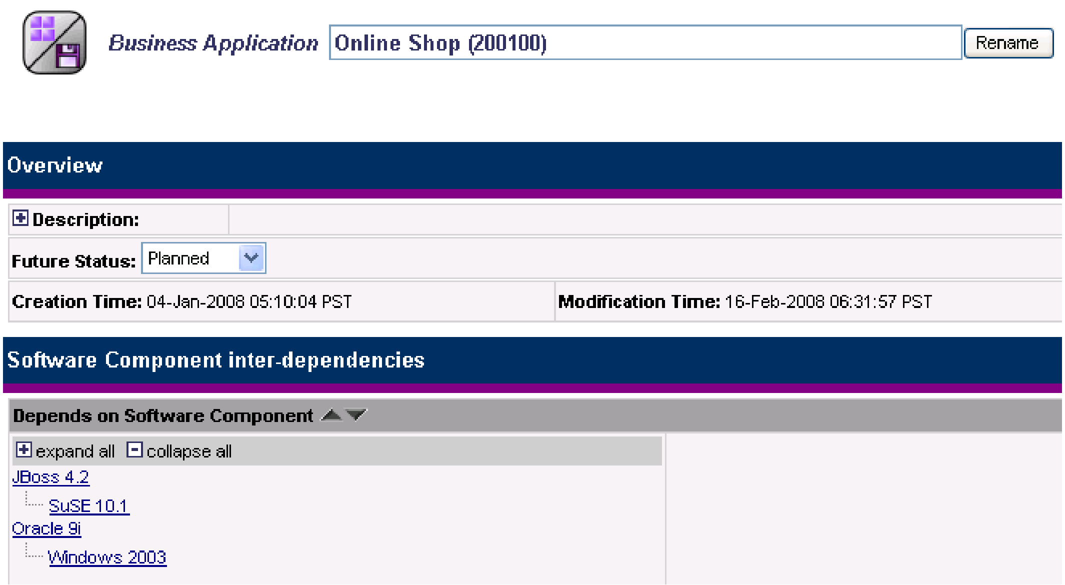


Figure 5.32.: *Adaptive EAM* - axis Infrastructure Management: Detail page of an application system showing the prerequisite infrastructure components

Determining the application systems, which are affected, if a specific infrastructure component, e. g. a database management system, should be retired, can be identified in the detail page of the respective component. A list of prerequisite infrastructure components of the current component is also given, allowing transitive analyses. Furthermore, using the RAM to create a query also taking into account lifecycle aspects associated to an infrastructure component (for a realization of lifecycle information see below), the user can create a tabular report on potentially affected application systems.

For performing a distinction of different types of infrastructure components, the functionality of user defined classification schemas can be used. Via this mechanism, it is possible to define an attribute for distinguishing, e. g. database management systems, application servers, and operating systems. If the individual infrastructure components are assigned such classifications, it is possible to utilize the advanced search for retrieving infrastructure components of a designated type. Figure 5.34 shows an example of a classification schema for infrastructure components.

As *Adaptive EAM* does not directly support lifecycle attributes for the infrastructure components, such information had to be incorporated making use of a custom classification schema. Thereby, the

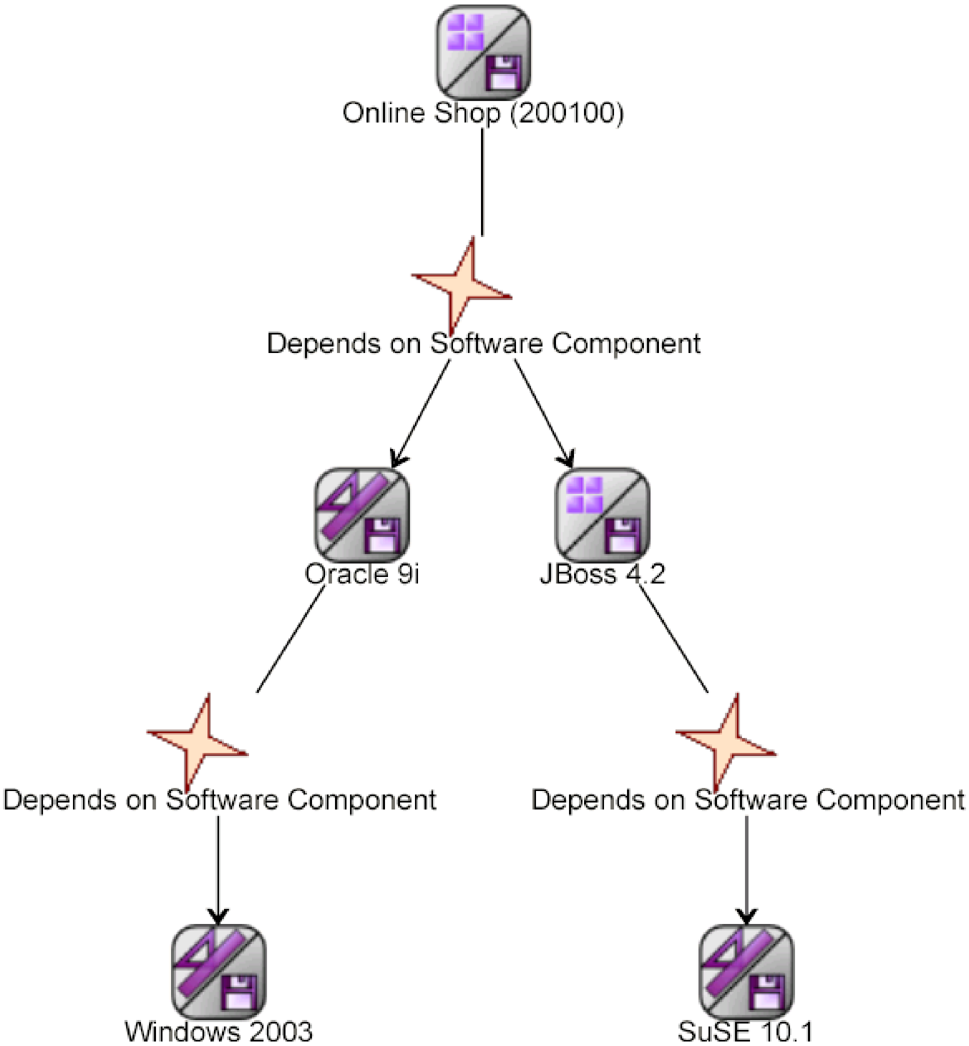


Figure 5.33.: *Adaptive EAM* - axis Infrastructure Management: Visualization of an application system together with corresponding infrastructure components

user could supply the vendors support periods and the *out of support date*. Such information could subsequently be used in performed advanced searches, but could not directly be visualized in *Adaptive EAM*, but via the integration to Microsoft Excel using the export of the embedded Microsoft Excel Web Component. Such lifecycle information could be populated into the repository utilizing the *Intelligent Classification Editor*, via which it would also be possible to keep this information for a large number of objects up to date.

Finally, in resprect to infrastructure management, the functionalities provided by the *Adaptive EAM discovery* tool have to be mentioned. By using this tool, it is possible to automatically search a network or parts thereof for running installations of certain infrastructure components, e. g. different versions of *Microsoft Windows*, of *Unix* or *Linux*, and of database management systems as the *Microsoft SQL server* or *Oracle*. The information automatically discovered can then be selected to be imported into the repository of *Adaptive EAM*, where further functionalityies, e. g. the rule based classification mechanisms could be leveraged, in order to complement the raw data with classifying information on lifecycle.

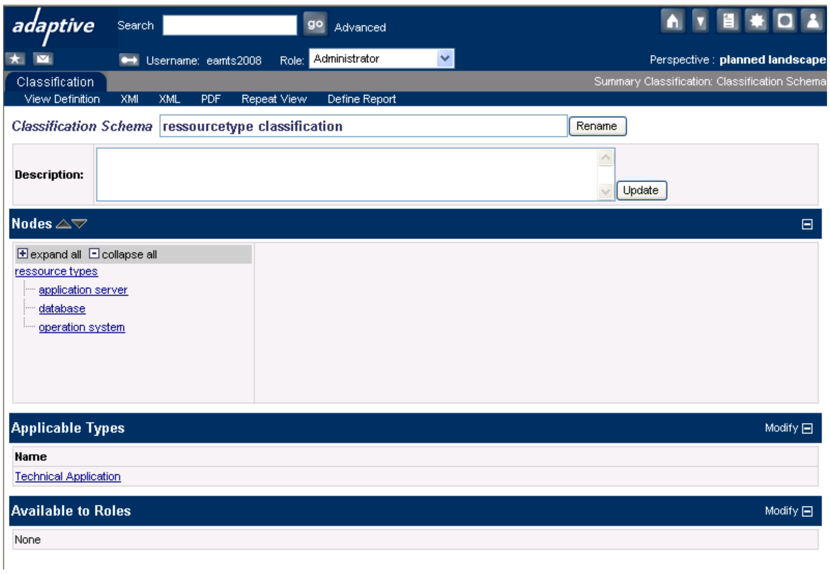


Figure 5.34.: *Adaptive EAM* - axis Infrastructure Management: Example of a classification schema

Procedure consistency is mostly given. Infrastructure components can be conveniently modeled, although e. g. lifecycle information is not supported out-of-the-box.

Procedure integration is mostly given. While information on application systems can be reused, a seamless integration to solution elements from IT architecture management is not directly provided.

Rating: 3

5.3. Tool Vendor's Profile

James Cerrato, Chief Product Officer
Pete Rivett, Chief Technology Officer, Member of the Architecture Board of the Object Management Group (OMG)

Adaptive is a global company, with headquarters in Virginia, USA and offices located in the United Kingdom, elsewhere in Europe and North America. Adaptive works with partners to enhance its offerings and to sell, deliver and support its suite of products in different geographies and vertical markets.

Adaptive's products and services enable organizations to maximize the value of their enterprise architecture initiatives, through exceptionally capable and feature-rich support for the full scope of Enterprise Architecture planning and implementation. By providing a big picture view - and with its unique capability to leverage disparate data sources into a holistic enterprise architecture framework - Adaptive provides management, in both business and IT, with the ability to plan, organize, lead and monitor change in a much more rigorous and effective manner.

Adaptive and Model Management

One of Adaptive's key distinctions is its emphasis on model management and indeed it is the only vendor focusing on model management today. This is achieved through an integrated web based environment for capturing, storing, visualizing, sharing, reusing and managing models in a single, reliable, secure and collaborative repository environment. Adaptive integrates with both modeling tools and Microsoft Office®tools. It has a web-based user interface for maintaining information where a specialist tool is not needed. Adaptive has a general and customizable capability applicable to any sort of model: and has used this customization capability to build a specific solution for Enterprise Architecture: Adaptive Enterprise Architecture Manager™.

Adaptive continues in invest key resources to ensure they are abreast of industry standards and best practices. This emphasis on standards allows Adaptive clients to maximize their investment in silo based proprietary data stores by providing a web based scalable solution which provides knowledge sharing, access control, analysis, reuse, workflow and collaboration on an enterprise-wide scale. As stated by Pete Rivett, Adaptive CTO, "a picture is worth a thousand words, and a model is worth a thousand pictures, Adaptive manages thousands of models."

The Adaptive Technology

Adaptive's technical architecture is 3 tier, using a web browser thin client and making use of best-of-breed application server and database technology for scalability. Through its active participation in standards bodies, Adaptive's metamodels, APIs and architecture are all fully standards-based.

Adaptive's application server uses common web standards where possible - including JSP, HTML, XML, XSL, SVG - in order to maximize customizablility at all levels. It has a separate tool for customization, Adaptive Designer™, which allows users to select objects, attributes and relationships from the metamodel to create logical views which are the basis of the browser views and reports: and will automatically generate the JSP and XSL code from these; the views can span information from many metamodels and tool sources to form 'big pictures' which can be associated with different user roles.

Adaptive has a very broad set of metamodels (over 900 classes in total), including many OMG standards, as well as Adaptive-specific metamodels. Its visualization capability supports both dynamically generated views (a model tells a thousand pictures) and has the ability to capture diagrams from modeling tools and render them, via a browser, without requiring the original tool.

Adaptive's products are enterprise-oriented which includes scalability, access control, and versioning, in addition to its collaboration capability amongst users via discussion groups, assessments and notifications.

Adaptive Tool Integrations

Adaptive Integrator™uses metamodeling standards to interoperate with over 50 tools, both to import, transform and integrate data and diagrams (through the use of the Diagram Interchange standard from OMG), and to export them. It provides full support for a number of out-of-the-box integrations.

In addition to the standard interfaces with these other tools, enabling Adaptive to rapidly adapt to support most flavors of XML or CSV, Adaptive Integrator can be configured to create custom imports, which can be used to very quickly adapt to variations in the level of tool support for XMI: these can be tied into a sophisticated Transformation Component infrastructure, which allows parameterized components (7 different types currently including XSL) to be pipelined into end-to-end transformation processes. Adaptive provides a Web Services API for integration purposes, in addition to browser-based import; and a remote command line (which allows for automatically scheduled imports).

Further, it provides an optional module called Adaptive Integrator Metadata Module which embeds technology from Meta Integration Technologies, Inc., a recognized leader in bridging to most leading modeling tools; Adaptive also includes capabilities for directly importing relational databases (mapped to metamodel).

Figure 5.35.: Logo of Adaptive

CHAPTER 6

alfabet AG (planningIT)

Contents

The alfabet AG provides the EA management tool *planningIT*, which is an integrated planning tool for IT management. The objective of *planningIT* is to support every business activity in the planning process from demand to budgeting. Thereby, the tool provides a repository with a given information model (called *Logical Inventory*), which builds the foundation for the different functionalities provided by the tool. These functionalities can be classified by the nine modules offered as shown in Figure 6.1.

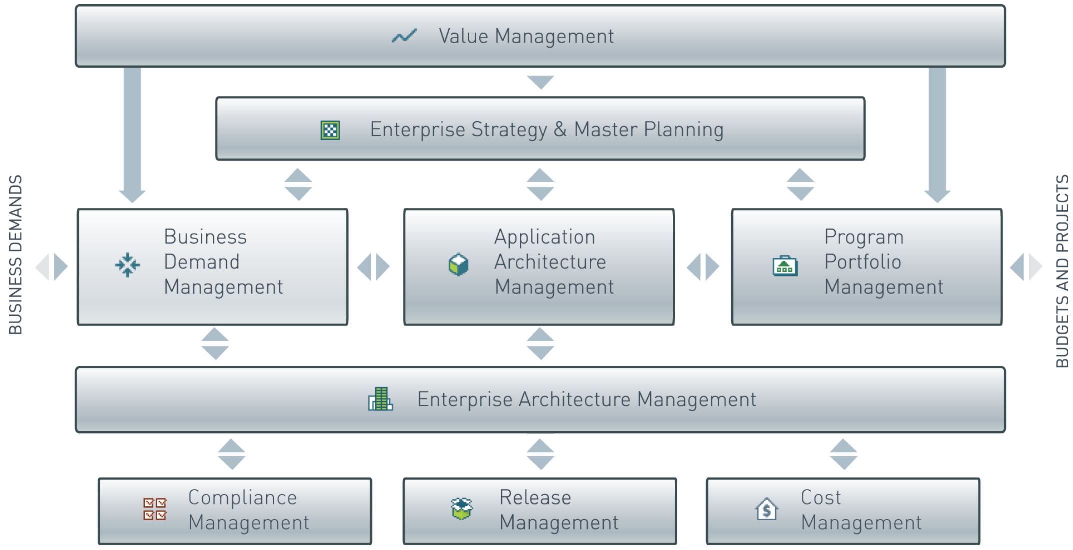


Figure 6.1.: Modules of *planningIT*

The version assessed in this survey is *planningIT* 3.1, the results of the evaluation are detailed below.

6.1. Evaluation of Specific Functionality

This section describes the results of the scenario simulation for specific functionality.

6.1.1. Importing, Editing, and Validating

Importing data into *planningIT* can be performed by using different import sources like Microsoft Excel, Microsoft Access, or XML. The setup for the initial import into *planningIT* is usually done by alfabet service and support. Thereby, the different concepts of the information model as outlined in Section 3.2 are supported and mapped to the given information model provided by *planningIT*. This transformation is usually done by mapping the structure of a Microsoft Access database to the corresponding elements of *planningIT*, thus, defining a mapping of classes and properties as well as a binding to create object relations.

planningIT does not provide sophisticated data transformation capabilities, as e. g. splitting comma separated values, as part of the mapping process. Such transformations are usually performed after the initial upload of data. Besides, data transformations can be performed using the built-in functionalities provided by Microsoft Excel, Microsoft Access, or scripting. Following the initial import of data into the *Logical Inventory*, external data sources can be integrated by using imports and exports realized with web services or via the product's XML-API. A Microsoft Word template approach is provided by *planningIT*, which can be used to gather updates to existing or newly-created data. Furthermore, *planningIT* provides import capabilities for different tools specialized in related domains such as the ARIS platform, SAP R/3, or HP Peregrine.

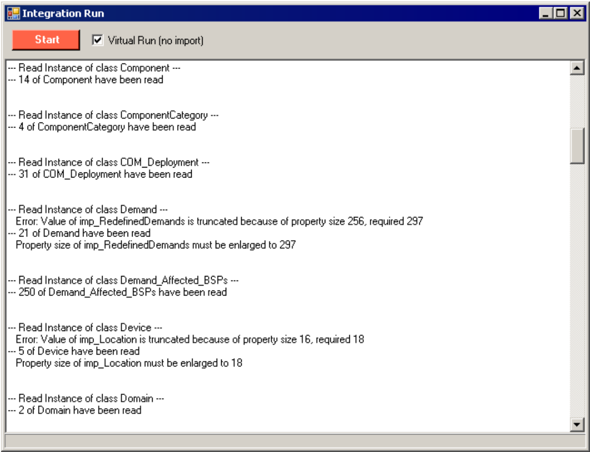


Figure 6.2.: *planningIT*- axis Importing, Editing, and Validating: Log file showing validation result during data import

The imported data is validated during the import process, e. g. according to aspects of type and length or the violation of standard or customer specific integrity or key constraints. Further consistency aspects, e. g. the maintenance of a mandatory attribute, can be implemented as part of the customer specific mapping or importing process. The validation information is provided in a log file, as shown in Figure 6.2.

Concerning editing functionalities, *planningIT* provides different supportive functions as for example wizards, editing of multiple data, pick lists for enumerations, and initializing of attributes with default values. In addition, even more sophisticated editing mechanisms as providing support for determining a priorizing attribute by performing multiple comparisons are provided by *planningIT* (e. g. used to determine the weights for *ValueNodes*).

If an external editor should be used to change the information maintained within the repository, *planningIT* provides different export functionalities varying from simple xml files to automatically generated Microsoft Word templates. Using the (re-)import functionalities of *planningIT* two different options are available merging data and importing new data. According to this distinction, a conflict resolution strategy is only necessary, if the merging functionality is used. A locking mechanism to avoid conflicts is supported by *planningIT* using the *Release Status* concept[1].

Rating: 6

6.1.2. Creating Visualizations

planningIT is shipped with support for numerous visualization types such as software maps, diagrams, and matrices illustrating different aspects of the EA. Thereby, the visualized symbols represent objects within the repository and are linked with the corresponding data, thus, changes in the repository are propagated automatically to the corresponding visualizations and vice versa.

Using the *alfabet Diagram Designer*, cluster maps (called *Standard Application Diagrams* within *planningIT*) can be created by clustering applications according to an *application group*[2]. Application groups can be defined by the user and provide functionalities for hierarchical grouping of applications. Concerning the requirements described in Section 4.1.2 a cluster map can be created by defining an

[1]This is for instance used for integration scenarios.
[2]This grouping can be done automatically during the import process.

application group that assorts the applications according to their deployment at a location, as shown in Figure 6.3. The *alfabet Diagram Designer* can be used to automatically load and arrange the associated applications of the application group.

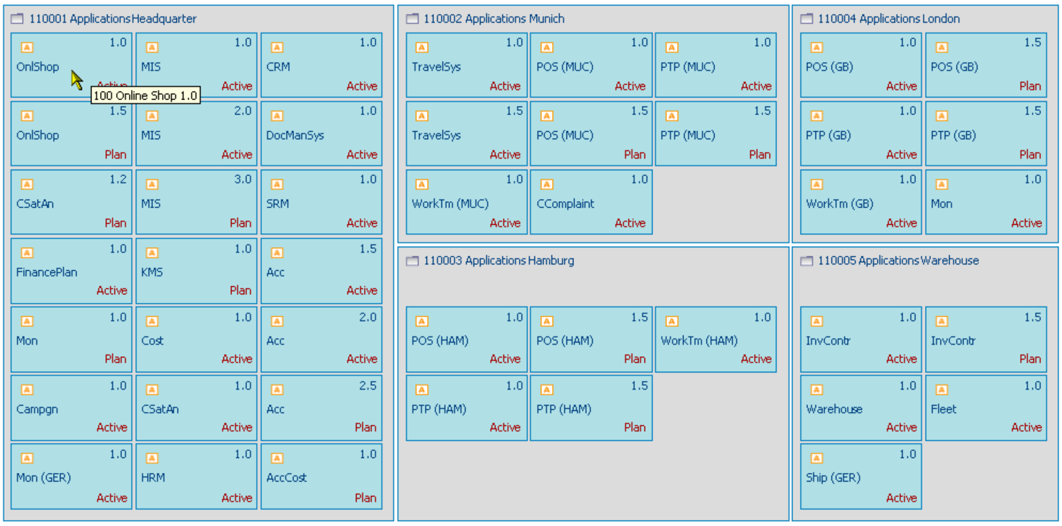


Figure 6.3.: *planningIT*- axis Creating Visualizations: Cluster map visualizing application groups and application systems

planningIT provides two different versions of process support maps (here called *Business Support Map*): an editable one, supporting graphical maintenance, as well as a read-only one (for examples see Figure 6.15 in Section 6.2.1). Both maps are generated automatically. Concerning the horizontal and vertical integration the read-only software map can be used to visualize integration by extending the graphical representation of an application system according to the provided business support over more than one business process or organizational unit (see Figures 6.14, 6.15, and Figure 6.16).

Time interval maps are called *Lifecycle* within *planningIT*(see Figure 6.4). Whereas, editing via dialogs and drill-down are supported on time interval maps, they do not support graphical editing. The visualizations are generated automatically out of data contained within the repository. Furthermore, an option to show the different versions of an application system on the map is provided, supporting various filtering options depending on the context.

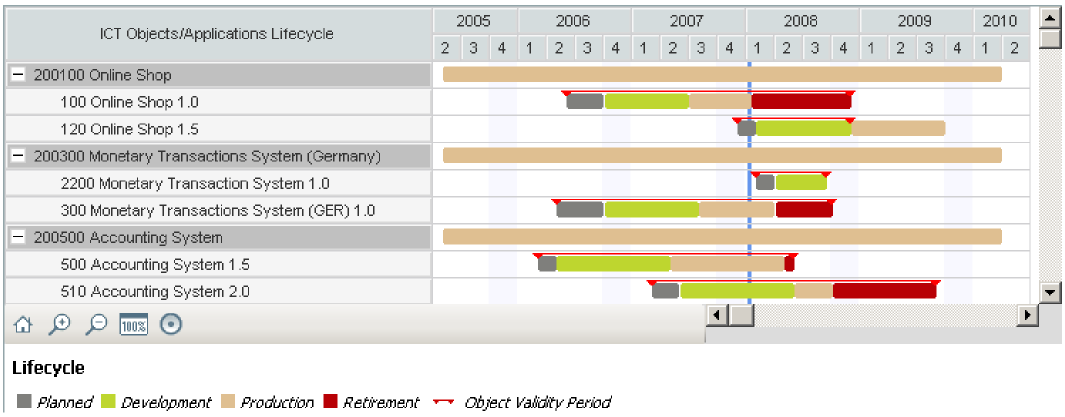


Figure 6.4.: *planningIT*- axis Creating Visualizations: Time interval map visualizing lifecycle information of application systems

planningIT supports the automated generation of graphlayout maps (called *Information Flow Diagrams* within *planningIT*), which show information flows between different applications as shown in Figure 6.5). Thereby, four different options to layout the map are offered by *planningIT* as e. g. Spring or Sugiyama layout. The visualization of different applications can be determined by the usage of several customizable filters (as e. g. screening for a certain objects state or active date).

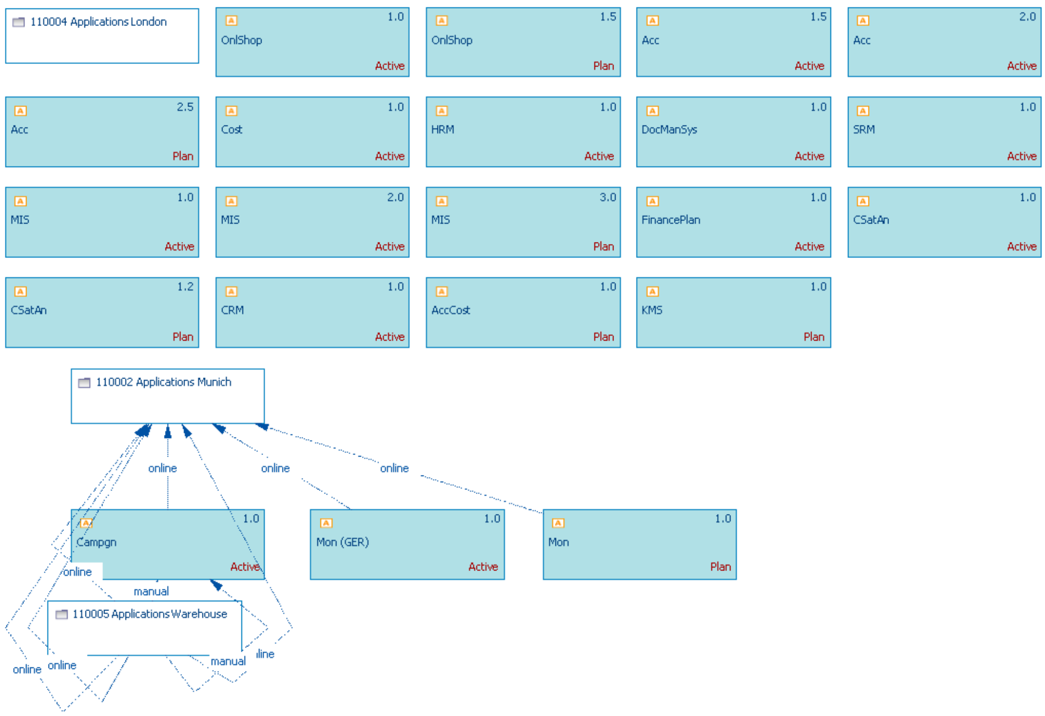


Figure 6.5.: *planningIT*- axis Creating Visualizations: Graphlayout map visualizing information flows

Swim lane diagrams are supported by *planningIT* in predefined contexts. For instance, a swim lane diagram called *Goal Dependency for Scorecard* (see Figure 6.6) is provided within the *Value* module of *planningIT*. This visualization can be used to illustrate the dependencies between different goals and to arrange these goals according to the typical perspectives (finances, customers, processes, and innovations). After the manual mapping of goals and perspectives, the corresponding read only diagram is generated automatically.

Portfolio matrices as the one shown in Figure 6.7 can be created in *planningIT* by configuring an *IT Portfolio*. A portfolio configuration consists of the definition of two or three axes. Thereby, the first two properties define the x- and y-axis of the portfolio matrix, while the optional third property is visualized using the size of the bubble. For the definition of the axis properties as well as calculations and/or scales of properties can be used. Concerning the visualization, the color of the bubbles is used to determine the name of the respective application system. The layout of the defined configuration is generated automatically and can be changed by editing the configuration.

Summarizing the capabilities provided by *planningIT* concerning the creation of visualizations it must be noted, that the provided functionalities focus on modeling instead of drawing. All diagrams mentioned before have a predefined association to the classes of *planningIT*, and the visualized objects always refer to an object maintained in the repository. In addition, design diagrams can be used to

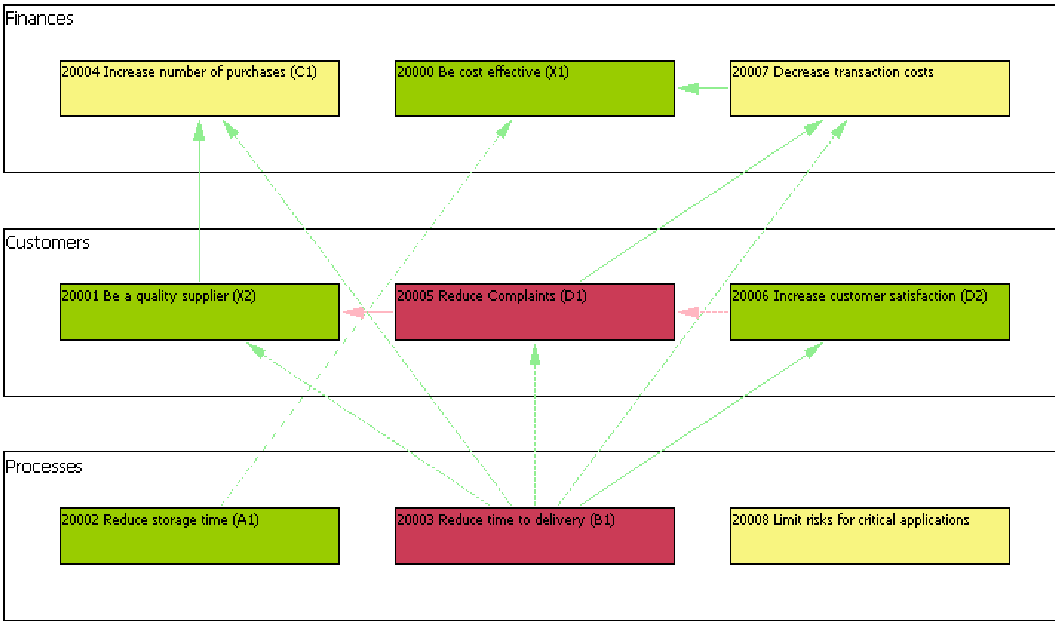


Figure 6.6.: *planningIT*- axis Creating Visualizations: Swim lane diagram visualizing the dependencies between different goals

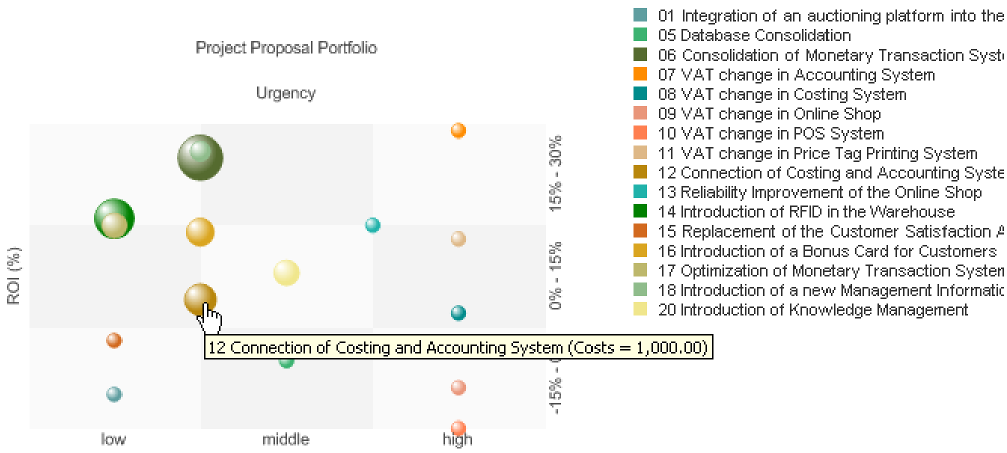


Figure 6.7.: *planningIT*- axis Creating Visualizations: Portfolio matrix illustrating different aspects of projects

visualize symbols without a semantic meaning (in terms of data maintained within the repository), which can be enhanced with symbols representing semantic information. The created diagrams and maps can be exported into several formats as e. g. PNG, SVG, or Microsoft PowerPoint as pictures.

Rating: 6

6.1.3. Interacting with, Editing of, and Annotating Visualizations

According to the approach of *planningIT* to support the planning process from demands to budgeting, visualizations are always created and used within a certain context. Thereby, all visualized objects are linked to the respective data contained within the repository. Thus, semantic changes performed by interactions with the visual representation of informations are reflected within the repository, and vice versa.

All the visualization types shipped with *planningIT* provide zooming and navigation mechanisms. The navigation mechanism supports double clicking on a visual symbol and jumping to the respective data set. Sole exception of this is the visual representation of a custom report as a Microsoft Office *PivotTable*, that does provide limited interaction capabilities, e. g. to change the arrangements of the rows and/or columns or calculation and sorting possibilities.

Using cluster maps or process support maps semantic changes on the respective data can be performed by interacting with the visual representation of an object as e. g. moving a rectangle, representing an application system, from the surrounding box representing an organizational unit to another organizational unit. By accomplishing this change the respective relationship between the objects in the repository is altered as well. In addition non semantic changes can be performed as e. g. changing the background color of a representation or resizing it. These non semantic changes, e. g. displaying the Human Resources System (700) always in the upper left corner of the organizational unit Munich are maintained during updates caused by data changes, such as updates caused by adding an information flow, but cannot be maintained by the tool during an automated regeneration of the visualization.

Whereas, filtering and drill-down functionalities are supported, the application portfolio diagram does not provide graphical interaction or editing capabilities. Changes can only be performed by changing the portfolio type, creating a new one, or changing the underlying data for the displayed objects.

planningIT provides multiple filtering mechanisms, e. g. highlighting objects of interest or hiding objects, which are not considered important by the user. Even more sophisticated interaction mechanisms are provided in predefined contexts as e. g. highlighting all elements that are connected to a selected one, for example via an interface, is e. g. supported in *Capability Architecture Diagrams*.

Besides the graphical representation of certain aspects, tabular reports such as the one shown in Figure 6.8 listing a user's application systems and user defined aspects thereof, are provided as part of the standard set of reports.

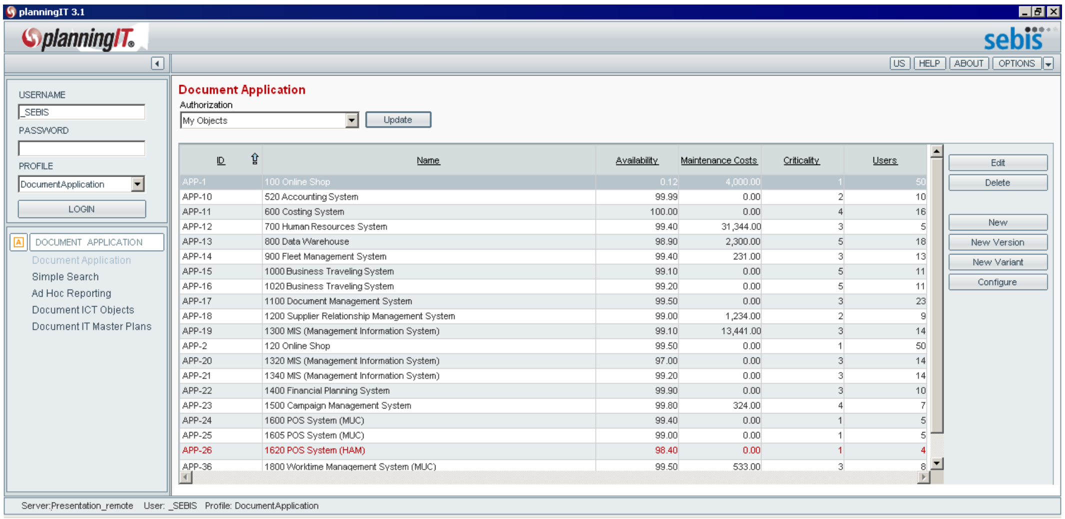


Figure 6.8.: *planningIT*- axis Interacting with, Editing of, and Annotating Visualizations: Screenshot of tabular report listing application systems and their attributes

Multiple aspects can be visualized in *planningIT* using annotation capabilities. An example is shown in Figure 6.9, where several aspects of application systems are visualized. Even though, *planningIT* does not provide the possibility to change the background color according to certain aspects, four different aspects can be visualized graphically and four other aspects can be annotated using alphanumerical representations at the same time per object type on a cluster map. A corresponding legend outlining the placement mapping for attributes and icons and the mapping of symbols used is generated automatically.

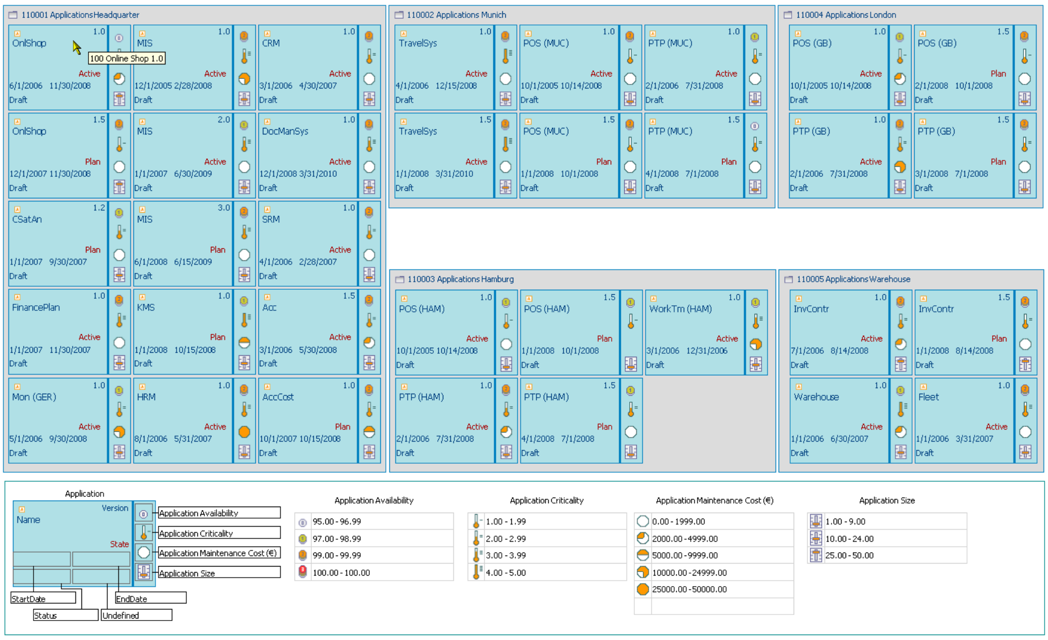


Figure 6.9.: *planningIT*- axis Interacting with, Editing of, and Annotating Visualizations: Screenshot of a cluster map annotated with multiple aspects

Rating: 6

6.1.4. Communication and Collaboration Support

planningIT is designed to support an integrated planning process within a collaborative environment with a large number of users. Thereby, different users and user groups can be created and defined using the administration profile. For every object contained within the repository of *planningIT* an authorized user and authorized user groups can be defined. Additionally, each object may refer to roles as description for personal or organizational responsibilities.

For providing a role-based access control, *planningIT* supports the definition of roles and profiles, that can be specified using an inheritance relationship. A role is defined as the functional relationship that a user or organization has to an object (including maps and diagrams). A profile defines the functional scope and the objects, which are accessible to the users with the matching user profile. Thus, a user may possess more than one user profile. Every profile owns a so called *Business Application* that defines the GUI layout for the respective profile. The GUI can be defined using *alfabet eXpand*, thereby determining the windows, dialogs, and information that are available within the profile.

Another possibility to organize and structure users and objects within *planningIT* is provided by the optional utilization of *mandates*. Mandates can be used to control the visibility of objects, thereby the accessibility of mandates takes precedence over the other concepts of access rights.

Concerning the type of access *planningIT* differentiates between read and read/write access. The so called *authorized user* or user group owns an object and has read/write access to the respective information. Besides, the authorized user can define a *deputy*, who possesses equal rights. If no authorized user is defined every user has read/write access. To restrict the accessibility of one or more objects a profile in read-only mode can be created.

Besides the authorization controls discussed above, *planningIT* provides for dynamically granted edit permissions to objects in the context of workflows and assignments. Users that have to make a contribution in one of the collaboration processes will be able to do so for the duration of the assigned task. The permissions are granted and revoked automatically.

For every object contained within the repository metadata is saved to enable auditing capabilities. Hence, *planningIT* provides the possibility to view the audit trail of any object, which contains e. g. creation date, last modification date, and an update history. Another possibility to keep track of changes is provided by *New or Recent Objects* that lists all the objects that have been recently modified or added. Furthermore, a configurable audit manager is provided, which allows fine granular tracking and reporting of changes according to the customers requirements. Figure 6.10 shows an example of an object history configured using the audit manager.

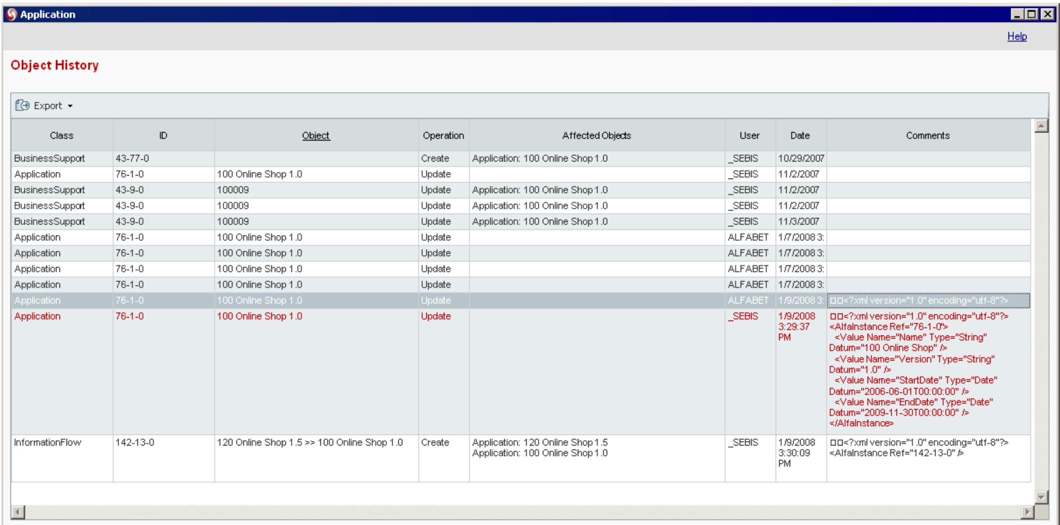


Figure 6.10.: *planningIT*- axis Communication and Collaboration Support: Object history of an application system

planningIT provides several possibilities to define different notification mechanisms using monitors. Thereby, *Activity*, *Inactivity*, and *Date Monitors* are distinguished. An activity monitor is used to receive a notification, if the monitored object is modified by another user. Monitoring of contextual changes such as changing an information flow or adding a demand, are supported. An inactivity monitor is typically used to ensure data is up-to-date. The owner of the monitor receives a notification, if the monitored object has not been changed within a specified period of time. Finally, the date monitor is typically used in the context of operationalization of plans. As, e. g. a lifecycle stage or release date for an application system approaches, the responsible user is reminded with a notification to confirm, change or reject current plans.

To support a collaborative environment and multiple users working together, *planningIT* provides the possibility to create *assignments* for an object[3]. An assignment belongs to a specific object and

[3] According to *alfabet*, with the upcoming release 4.0 a highly configurable automated workflow management functionality is going to be provided.

must contain a title as well as a target date by which it should be completed. Finally, it must be assigned to a user, that will receive an e-mail notification informing him about the assignment. Additionally, an assignment can be marked as optional or mandatory and have a status. Much like ordinary e-mail systems *planningIT*'s user interface provides an in- and out-box for assignments. Also, the assignments are accessible from the object they pertain to. Furthermore, *planningIT* allows for notifications to be generated by users triggering the review or revision of information they find incorrect, incomplete or doubtful. Further facilitating the collaborative discussion process a notepad functionality is provided.

Supporting the multiuser environment, *planningIT* provides alongside with the desktop solution realized as slim client, a thin client solution that can be accessed via the web using Microsoft Internet Explorer to ensure the enterprise-wide visibility of the EA management initiative. The functionality of the slim client and the thin client is nearly identical, except for the *alfabet Diagram Designer*, which provides design capabilities for manually designed diagrams and is only available within the slim client.

Rating: 7

6.1.5. Flexibility of the Information Model

planningIT focuses on the support of the planning process from demand to budget. Thus, it provides a strong process guidance, which is based on a predefined information model, containing classes, attributes, and relationships. *planningIT* does hence not follow the metamodeling approach. Accordingly, the flexibility of the information model and its adaptability are limited as it may be amended but not structurally altered. The customer can exclude but not delete certain elements of the information model ensuring consistency with the methodology[4].

If a mapping between the enterprise-specific information model, as e. g. supplied in this survey, and the information model provided by *planningIT* is necessary, the mapping would be usually done by alfabet certified consultants. Although, using the tool *alfabet eXpand* changes to the information model can be performed directly by the user, e. g. the introduction, modification, and deletion of user-defined classes, attributes, or relationships.

The information model provided by *planningIT* is designed in an object oriented way. In addition to the general concepts of classes, attributes, and relationships more sophisticated concepts like different multiplicities (0, 1, 0..*, 0..1, n..m) for relationship ends, default values for attributes, and optional vs. mandatory attributes are supported. *planningIT* provides attributes with predefined types (e. g. String, Boolean, Date, Integer, Reference, Reference Array) as well as customer defined enumerations.

Changes to the information model are made visible within the graphical user interface of *planningIT*, by defining custom views for the newly introduced classes and attributes. The configuration can also be used to hide attributes or classes from the predefined information model either completely or for individual profiles.

Rating: 4

6.1.6. Support of large scale Data

Importing large scale data into *planningIT*, for example of up to 10000 application systems, does not cause any problems. Within the different modules of *planningIT*, e. g. the *Inventory* module,

[4] According to alfabet, this approach permits the possibility for the user to latter reconsider such decisions and use further parts of the provided methodology.

the handling is not visibly affected by the number of artifacts contained in the repository. Creating, reading, updating, or deleting an artifact can be performed in real-time.

planningIT provides in fact some supporting functionalities to deal with large scale data, as e. g. the grouping of elements alphabetically to ease the navigation within the explorer view. Another example for a supportive functionality of *planningIT* in the context of large scale data is contained within the *Search* module, where a search for elements can be filtered by using a pattern-based search string.

Concerning the creation of visualizations a drawback in the performance must be noted, if a large visualization is created automatically, e. g. a cluster map containing all application systems, their hosting relationship to an organizational unit, and their interconnections. Nevertheless, the interaction capabilities of visualizations are not affected by the number of application systems unlike the loading of a saved visualization, which takes considerably longer if thousands of application systems with tens or hundreds of thousand interconnections must be visualized.

Mastering the scaling challenge is possible by aggregating information in the diagram combined with drill-down navigation. Based on an aggregation of application systems into an hierarchy of application groups an aggregation of information flows from application system level to application group level is performed, as visualized in Figure 6.11. For these aggregating cluster map and the associated drill-down interactions performance degradation is not noticeable.

Rating: 7

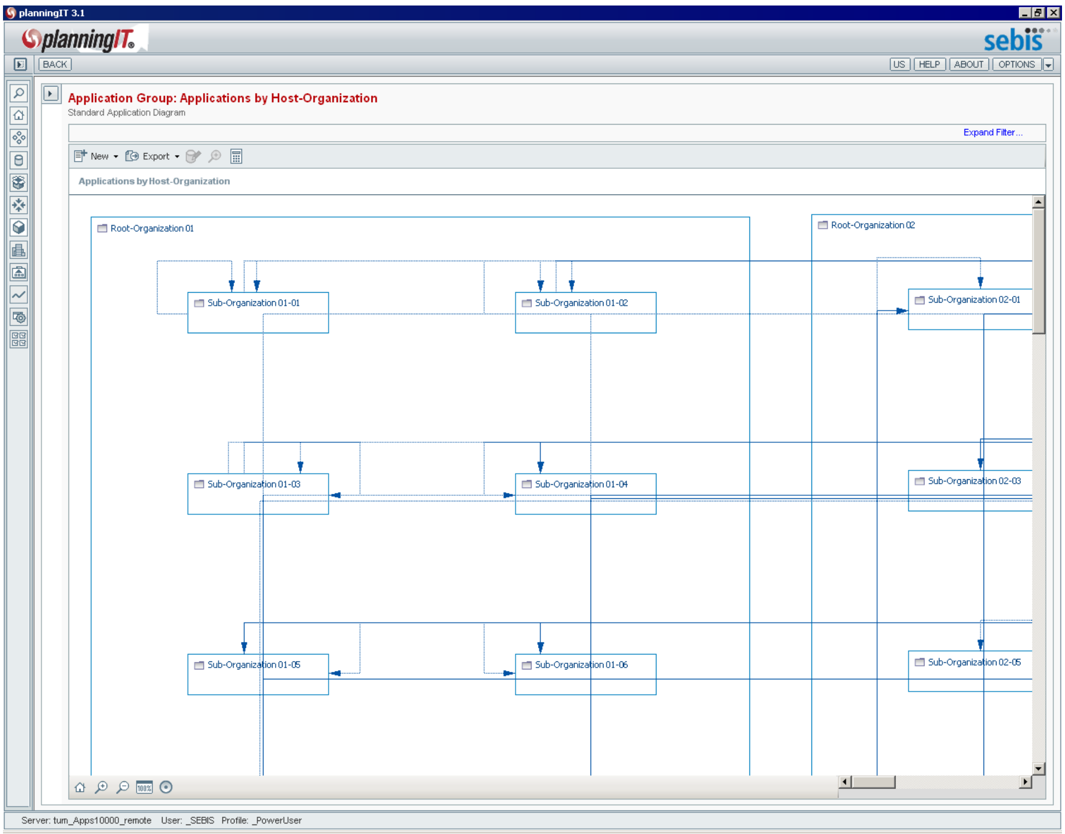


Figure 6.11.: *planningIT*- axis Support of large scale Data: Screenshot of an aggregating cluster map

6.1.7. Impact Analysis and Reporting

planningIT comprises a rich set of predefined reports supporting the analysis of the EA, e. g. affected architecture elements in respect to a given demand or project proposal. Figure 6.12 visualizes architecture elements that are affected by an ongoing project. In addition to these predefined reports, *Custom Reports* can be defined, that support impact analysis computations based on the information stored within the repository.

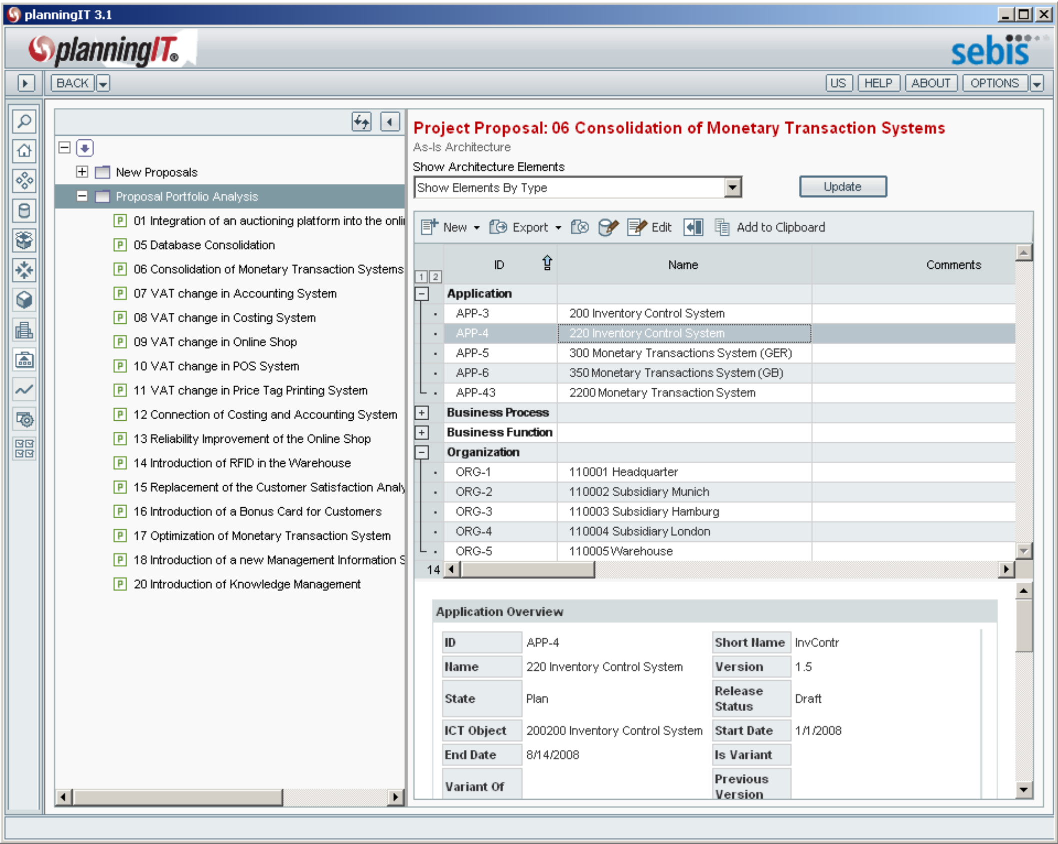


Figure 6.12.: *planningIT*- axis Impact Analysis and Reporting: Standard report visualizing affected architecture elements of a selected project proposal

Using the *alfabet eXpand* tool, custom reports can be defined with a descriptive UI, the *alfabet Query Builder*, as illustrated in Figure 6.13. The visualized report lists the business processes and the number of application systems supporting them. The *alfabet eXpand* tool provides an expressive query language, providing different types of joins (inner, left, right, and full) using compare operations (equal to, lesser than, greater than, ...) on the selection of elements. Furthermore, aggregation functions as e. g. count, sum, average are provided. The created custom reports are accessible for users, according to their access rights on the selected base class of the report. All generated custom reports may optionally be rendered as Microsoft Office *Pivot Table*.

Furthermore, a possibility for impact analysis is provided using *prioritization schemes*, which can be used to define the content of portfolio axes. Thus, graphical visualizations can be used to highlight objects due to an impact analysis. In the exemplary portfolio matrix shown in Figure 6.7 project proposals are visualized with their expected return on investment (ROI) on the y-axis and their urgency on the x-axis. In addition, the project costs have been defined as prioritization scheme mapped to the size of the bubbles.

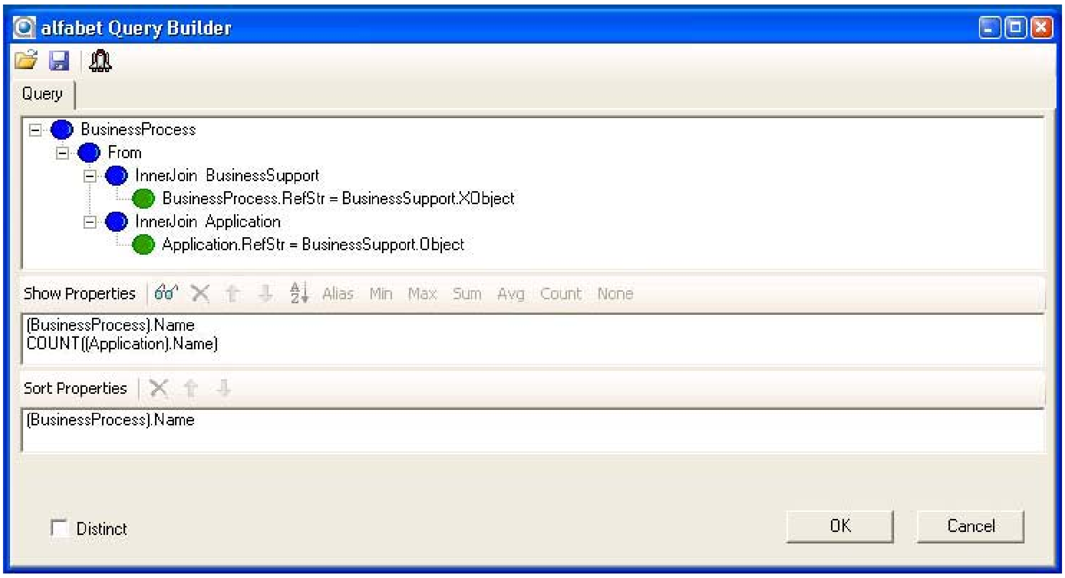


Figure 6.13.: *planningIT*- axis Impact Analysis and Reporting: Query builder used for the creation of a custom report

Other possibilities provided by *planningIT* to support impact analysis can be found using graphlayout maps, which can be used to analyze information flows between different application systems as well as swim lane diagrams, which support the dependency analysis of goals. Thereby, these diagrams are associated with predefined contexts based on the process framework and methodology supported by *planningIT*. All reports may directly be exported from the user interface for further processing in MS Office environments.

Rating: 7

6.1.8. Usability

The standard menu structure of *planningIT* is derived from the different modules as visualized in Figure 6.1. These nine modules of EA management tasks are complemented by the additional menu entries *Search, Home,* and *Config*. Furthermore, the menu entry *Inventory* allows the user to access the repository, which is subdivided into the areas *Business Architecture, Information Architecture, Application Architecture, Technical Architecture*, and *Deployment Architecture*. Each of these substructure navigation elements provides functionalities to create, update, or delete the corresponding artifacts of the EA contained within the repository. The substructure of the other modules are built task oriented, e. g. the module *Enterprise* contains the item *Plan Target Architecture* for planning the target landscape.

Being a tool providing support for EA management, *planningIT* needs to satisfy the requirements of many different stakeholders. On the one hand, everyday users such as enterprise architects, landscape planners, or IT strategists make use of multiple functionalities provided by *planningIT*. After familiarization, they will experience no problems navigating through the menus and finding specific functionalities. On the other hand, the EA management process requires continuous contributions from occasional users, like application owners, project managers, or business process owners, which will use only a small subset of the functionality and who do not want to deal with the multitude of options provided the tool.

Therefore, *planningIT* provides features to support tailoring the GUI according to the user's need. Leveraging the profile mechanism, the appearance of *planningIT* can be configured for easy use. For an example see Figure 6.8, where a user is provided with a simple menu structure according to the selected profile "DocumentApplication". Thus, allowing the user to accomplish the EA specific duties of their role according to their tasks as defined within the EA management process.

planningIT supports users in multiple ways during the execution of their tasks. HTML-based guide pages can be associated to each of the modules of a profile, allowing the user to easily navigate to the right location as the guide pages can link to *planningIT*'s functionality. Thereby, customers are provided a mechanism to customize the GUI according to their corporate design and the specific need of a group of stakeholders. Also, wizards can be configured according to customer needs, which guide the user through the single steps of a defined process. Further support is delivered by the search navigation, which provides the possibility to use pattern-based expressions. Finally, a *back* button is provided to easily navigate to previously used views.

In addition to the role-based appearance of the tool, *planningIT* provides the possibility to use dashboards and bookmarks. Dashboards provide a portal-like environment that allows users easy navigation. A dashboard can contain text and bookmarks to allow quick navigation to areas of regular or present interest. For example an application landscape planner could create himself a dashboard containing bookmarks for a visualization of the current, planned, and target landscapes.

Taking all this into account, some training to comprehend all functionalities offered by *planningIT* would be advisable before using it for the first time. Nevertheless, after a period of familiarization the basic and advanced functionalities can be used by an ordinary user. Thereby, the documentation of *planningIT*, which covers an index structure, a glossary, and a user manual designed as tutorial provides a good starting point to understand the modules and the functionalities provided by *planningIT*.

Rating: 6

6.2. Evaluation of EA Management Support

This section describes the results of the scenario simulation for EA management support.

6.2.1. Landscape Management

The *planningIT* module *Enterprise Architecture Management* provides functionalities to plan, analyze, and control the evolution of the application landscape. Thereby, *planningIT* differs between the *as-is architecture*, called current landscape in our terms, and *to-be architectures*, which are subdivided into *medium-term* (planned landscapes) and *long-term architectures* (target landscapes).

In addition, *planningIT* uses the concept of so called *ICT*[5] *Objects*, which constitute an abstract object, that represents an architectural element (e. g. application, component, or standard platform) regardless of its versioning. An ICT Object can be used e. g. within a planned or target application landscape to define a business process support provided by a system without knowing the exact version.

Furthermore, *planningIT* differentiates between four different kinds of business support: the *operational business support* describes an active or firmly planned support provided by an application system; the *solution business support* defines a proposed business support in the context of an architecture solution defined within a project proposal; the *strategic business support* represents the to-be

[5]ICT is used in *planningIT* as abbreviation of *Information and Communication Technology.*

business support in the target landscape that is based on an ICT Object; the *tactical business support* represents support in the context of a planned landscape, based on an application system or an ICT Object.

planningIT uses the concept of *master plans*, which describe the entirety of tactical business support, to plan the evolution of the application landscape. A master plan in *planningIT* is broken out into multiple *master plan maps*, that detail the planned support for business processes in organizations by existing or planned application systems or ICT Objects within a specific time frame. This way master plans are made manageable in federated, complex or heterogeneous environments. Figure 6.14 visualizes a process support map, detailing the planned evolution of the application landscape, until December 2010. If changes are applied to a business support, which is displayed on one of the other maps of the master plan, there each map belonging to a master plan is cross synchronized.

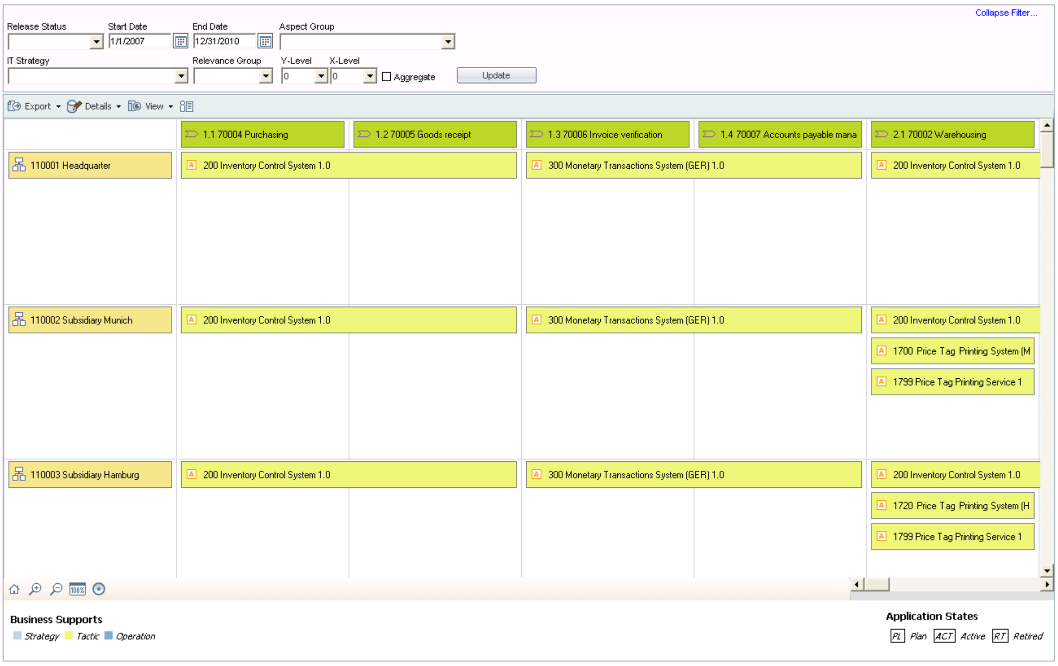


Figure 6.14.: *planningIT*- axis Landscape Management: Process support map visualizing the planned landscape

The landscape planning within *planningIT* is based on process support maps. Graphlayout maps can be used in the same way. Using process support maps to manage the evolution of the landscape, the planned landscape can graphically be modeled by selecting the organizational units and business processes for which the evolution should be planned. *planningIT* supports the automated loading of application systems, that currently provide the business support. The elements of the business support map, e. g. tactical business supports, can afterwards be created, modified, or deleted manually by the user.

Visualizing the different status of the evolution can be done by using the filtering mechanism provided by *planningIT*. The current landscape for example is shown by filtering the visualization of master map elements according to the respective time period. The screenshot in Figure 6.15 visualizes the current landscape of *SoCaStore*.

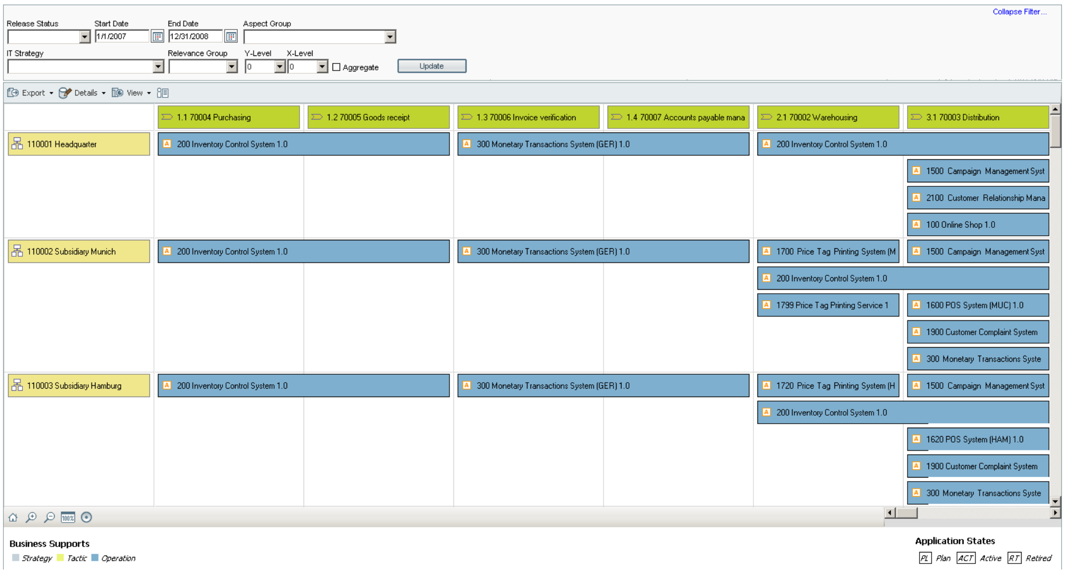


Figure 6.15.: *planningIT*- axis Landscape Management: Process support map visualizing the current landscape

The target landscape is created using ICT Objects[6], representing strategic business supports. The target landscape can be used to evaluate IT master plans concerning their accordance to the target landscape. Figure 6.16 shows a screenshot of a business analysis visualizing the current, planned, and target landscape with vertical integration.

Comparisons between two planned landscapes can be visualized using a *Comparison Report* as shown in Figure 6.17. This report contains three categories, *Common to both Master Plans* and one for each of the two compared master plans, thereby, providing a convenient overview of the differences. Facilitating traceability, *planningIT* provides the possibility to store a comment for a change in the description of a support relationship.

Due to the fact that all information displayed on process support maps within *planningIT* is directly loaded from the repository, planned landscapes, that are outdated, are typically not saved. Therefore, the graphical monitoring of changes in planned landscapes is not directly supported. Using distinct master plans for updates on the planned landscape, a comparison between the planned landscape for the 01-01-2008 made in January 2007 and in July 2007 can be done. Otherwise, the respective master plan map would have to be archived externally.

The procedure consistency is completely given. *planningIT* provides a very similar procedure within the *Enterprise* module as the one described in the scenario documentation (see Chapter 4).

The procedure integration is completely given, e. g. lifecycle information of application systems can be visualized on process support maps. Additionally, already defined planned business supports can be used to initialize the planned landscape.

Rating: 7

[6] ICT Objects act as an abstraction or container for e. g. applications, components, or standard platforms and are not versioned to support evolution planning without knowing the specific element that will be used.

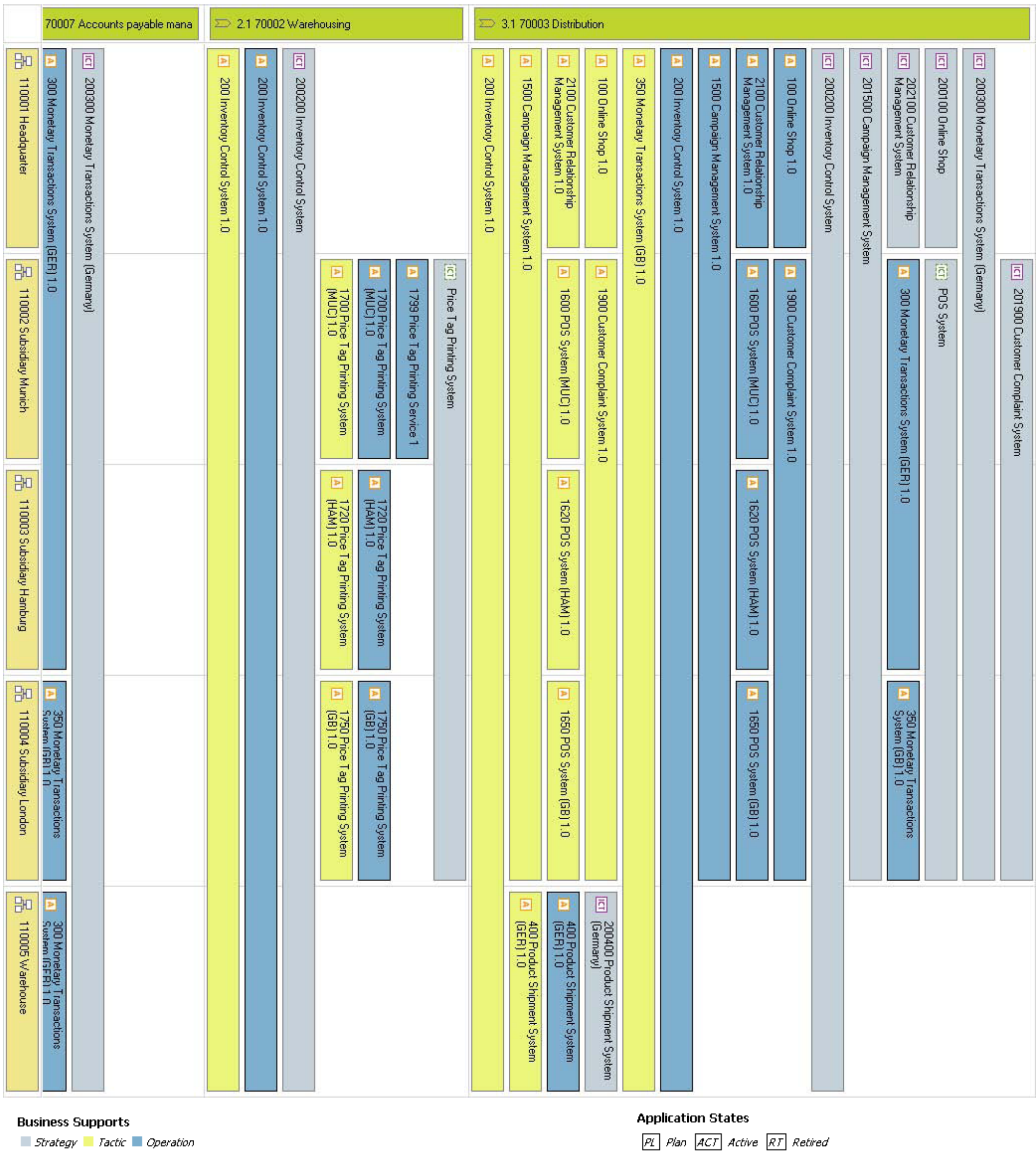


Figure 6.16.: *planningIT*- axis Landscape Management: Process support map visualizing the target landscape using vertical integration

6.2.2. Demand Management

Using the *Demand* module of *planningIT*, different functionalities for viewing, analyzing, controlling, and aligning requests for IT supports with business priorities and strategic goals are provided. The module is substructured according to the different tasks performed within the demand management process. Entering a demand can be done using the *Capture Demand* menu entry. Thereby, the user can either edit a previously defined demand, e. g. after using the multi-dimensional search mechanism, or create a new demand.

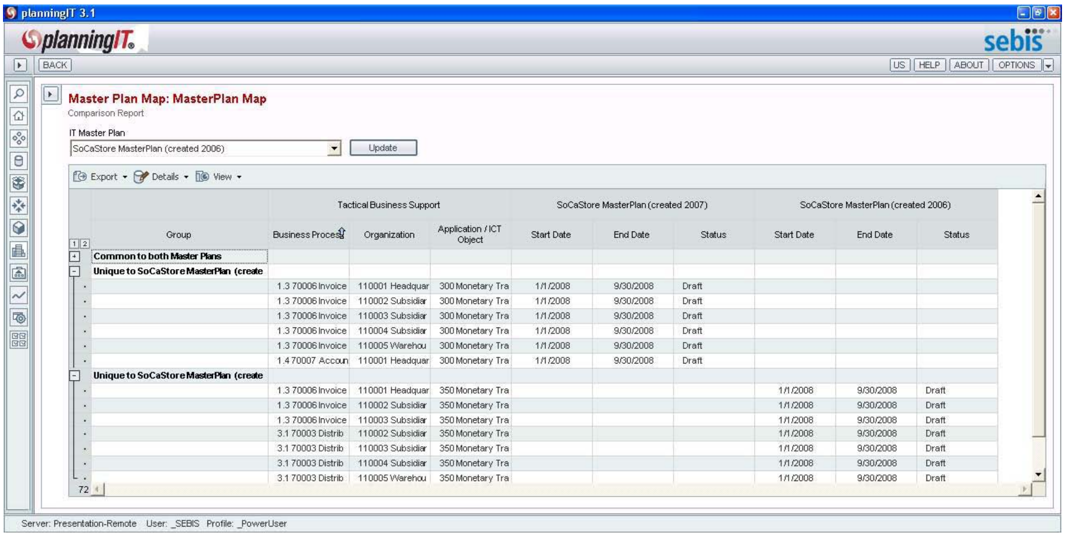


Figure 6.17.: *planningIT*- axis Landscape Management: Comparison report between two planned landscapes

All received demands, including their attributes, can be viewed within the demand management menu entry as visualized in Figure 6.18. Thereby, *planningIT* provides multiple filtering mechanisms, e. g. screening all demands requested from a specific organizational unit, a given status, or classification. For every demand the strategic alignment can be described by defining the goal(s) that the selected demand helps achieve.

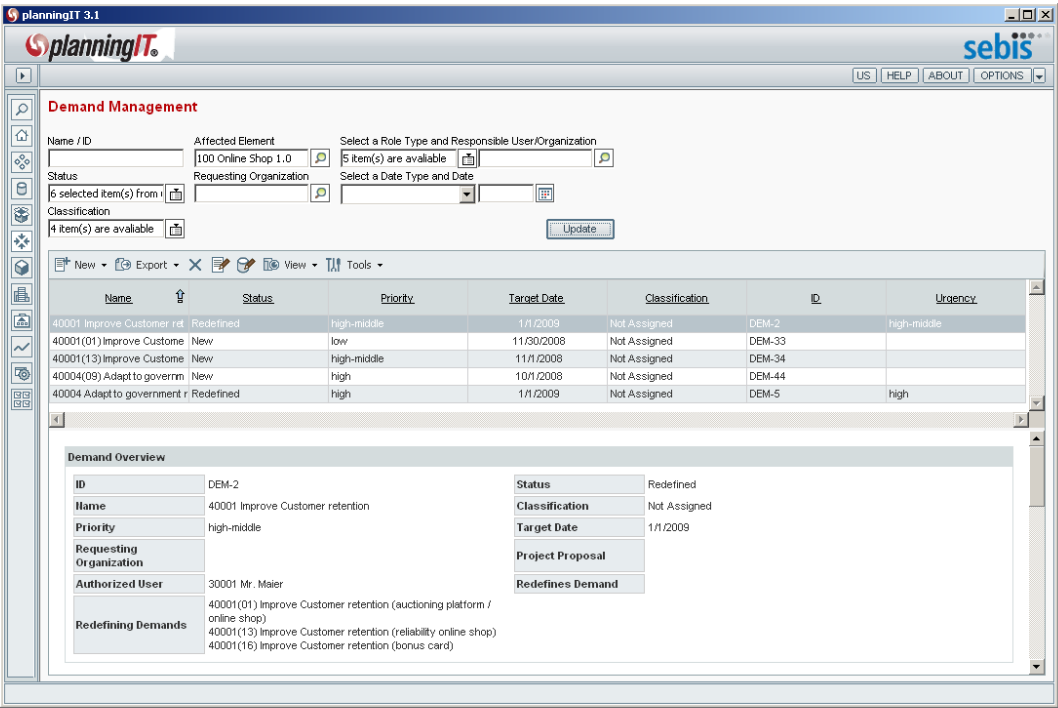


Figure 6.18.: *planningIT*- axis Demand Management: List of received demands that affect a specified application system

Further, affected architectural elements like application systems, business processes, business functions, etc. can be associated with one or more demands. A standard report listing the demands' affected architectural elements is available (cf. Figure 6.19). Following the integrated approach of *planningIT*, traversing from the organizational unit via the requested demands back to the affected architectural elements is possible.

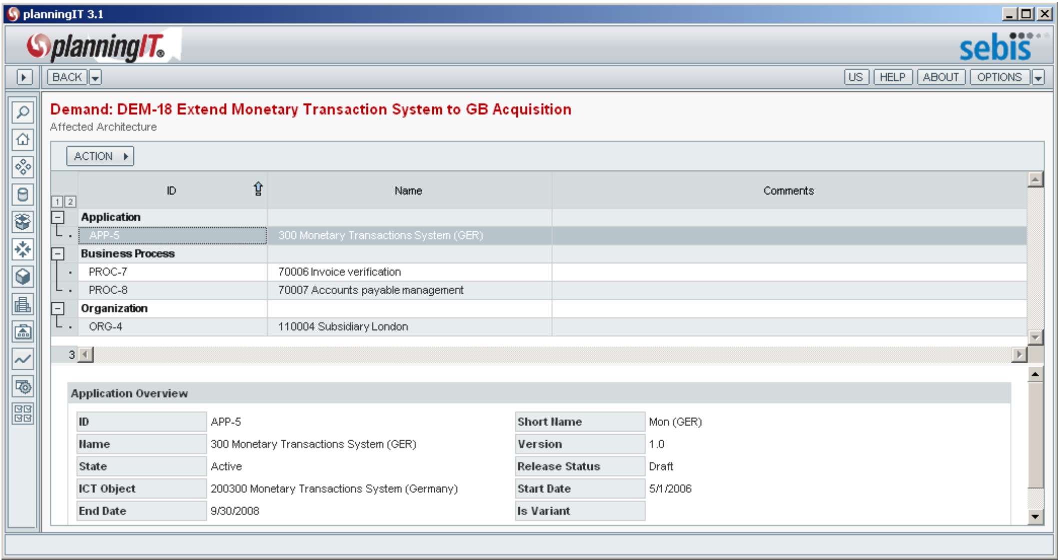


Figure 6.19.: *planningIT*- axis Demand Management: Report listing affected architecture elements

One or more demands can result in a project proposal. The *Project Proposal Management* view lists all project proposals currently defined. To assure end-to-end demand tracking *planningIT* supports a mapping of demands to at most one project proposal. However, *planningIT* supports the split up of demands that are more encompassing into smaller ones using an in-built demand refinement process first and mapping the resulting demands to different project proposals afterwards. Furthermore, *planningIT* supports the demand management process by providing analysis support as e. g. automatically identifying similar project proposals for a given project proposal based on overlaps in the affected architectural elements or visualizing portfolios of projects that share affected architecture elements with the selected one.

The procedure consistency is completely given. *planningIT* provides a dedicated module (*Demand*) to handle demands. A graphical analysis as introduced in Section 4.2.2 is not directly supported. Equivalent information can easily be provided in a custom report using *planningIT*'s object query language.

The procedure integration is completely given, e. g. impact analysis on the demands can be performed to identify the affected architectural elements. Furthermore, demand portfolio analysis are well supported by *planningIT*, so the user can identify similar demands that could be combined in one project proposal.

Rating: 7

6.2.3. Project Portfolio Management

planningIT provides an overview about the existing project proposals within the *Portfolio* module similar to the overview provided in the *Demand* module as described in Section 6.2.2. Thereby, for every project proposal different attributes (as e. g. name, planned start/end date, urgency) and relationships (as e. g. the affected architectural elements) are maintained either automatically through the assigned demands of Section 6.2.2 or manually by the user. Financial analysis for project proposals, is supported via business cases with automated calculations of configurable key performance metrics based on configurable cost and benefit types.

planningIT provides analysis of the project portfolio in multiple ways, e. g. via automated generation of a portfolio matrix visualizing similar project portfolios in respect to the selected one or creating Gantt diagrams illustrating project proposals, which overlap on the adressed architecture elements. Figure 6.20 shows a portfolio matrix illustrating the urgency of a project proposal on the x-axis, the expected ROI on the y-axis, and the strategic impact of an project proposal as the size of the bubbles.

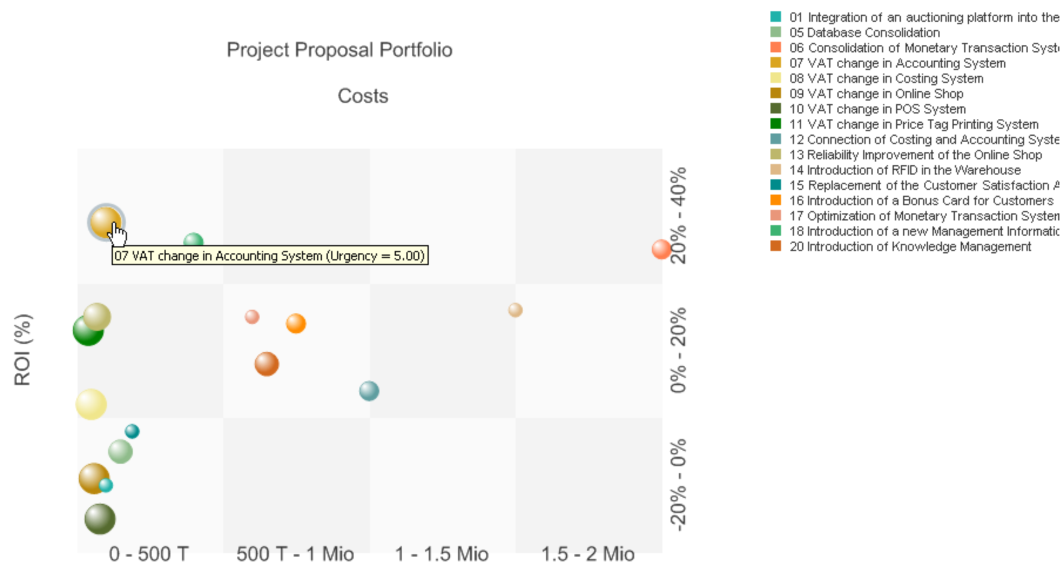


Figure 6.20.: *planningIT*- axis Project Portfolio Management: Portfolio matrix showing the urgency, expected ROI, and size of a project proposal

Using the *To-Be Architecture Definition* and *Analysis* functionality provided by *planningIT* the changes performed on architectural elements affected by the project proposal can be defined, reviewed, or analyzed. Thereby, graphical representations, as e. g. a process support maps, which illustrates one solution[7] and the affected elements are supported (cf. Figure 6.21).

Furthermore, *planningIT* supports the prioritization of project proposals according to user defined prioritization schemes. Figure 6.22 gives an overview about projects and lists them according to the calculated ranking (in the example the rating is calculated according to the strategic impact rating and the urgency of the projects each aspect weighted with 50%). Thereby, a distinction is made between projects that have to be performed because of legal guidelines, projects that are currently running, and a third category containing all other projects. The automatically generated ranking can then be adjusted manually.

The procedure consistency is mostly given. Two of the objectives (e. g. see Figure 4.23 or Figure 4.24) are not available out of the box, but can be easily achieved through a custom report.

[7]A *solution* in *planningIT* describes an alternative planned architecture that is designed to realize a project proposal.

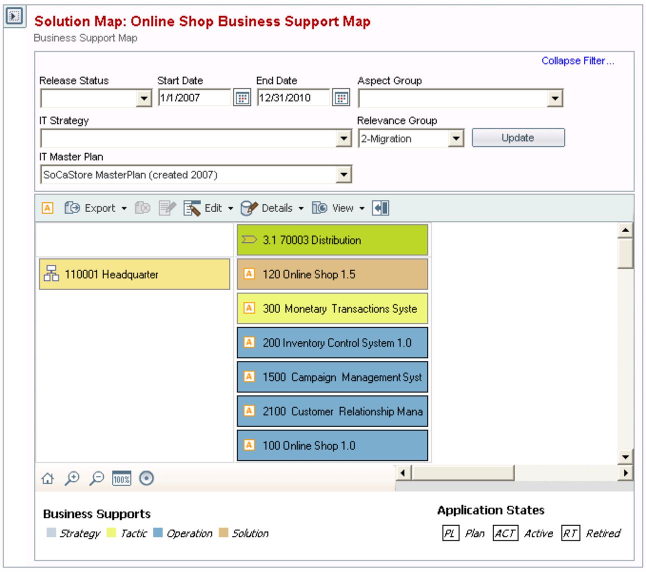


Figure 6.21.: *planningIT*- axis Project Portfolio Management: Process support map illustrating the business impact of a solution

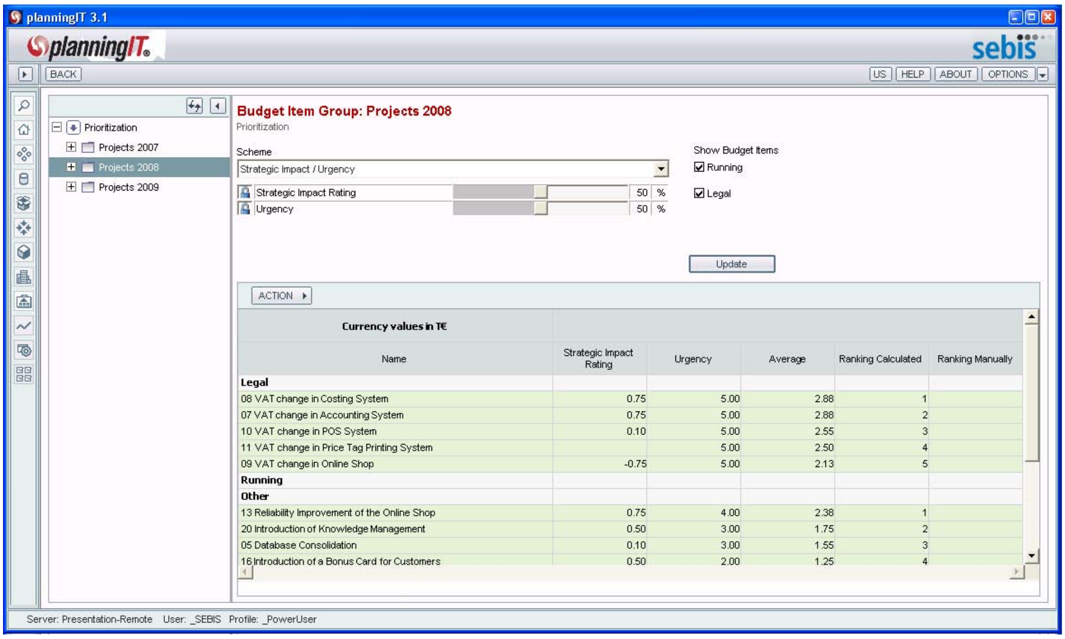


Figure 6.22.: *planningIT*- axis Project Portfolio Management: List of projects ordered by their calculated ranking according to the given parameters

The procedure integration is completely given. *planningIT* provides a tight integration to other EA management tasks as e. g. functionality is provided to define work items and a time schedule for a project.

Rating: 6

6.2.4. Synchronization Management

planningIT provides only limited support for synchronization management within the *Portfolio* module[8]. Impact analysis across multiple (or all) projects and their affected architecture elements, as e.g. application systems, organizational units, process supports, as requested with Figure 4.30 is not directly supported by *planningIT* but can be easily introduced by leveraging the custom report functionalities[9]. Following the integrated approach a *Change Request Analysis*, as required by this scenario, can be performed on different architectural elements (e.g. application systems) listing the relevant demands and project proposals that change the selected element.

Furthermore, *planningIT* provides the possibility to conduct a *Dependent Projects Gantt* analysis as visualized in Figure 6.23. Thereby, different affected elements can be selected and combined to form the basis of the dependency analysis. Although, project proposals have a planned start and end date, management of project delays is currently not supported in *planningIT*. Therefore, changing the planned end date of a project proposal does not cause any automated notification in case of potential knock-on effects to other projects caused by the delay. However, using the dependent project diagram as visualized in Figure 6.23 the user is supported in identifying candidate projects that could be affected by a delay. The overlap of the two project proposals in the first half of October 2008 caused by a delay of project proposal 12, for instance, can be viewed as causing a potential knock-on to project proposal 07.

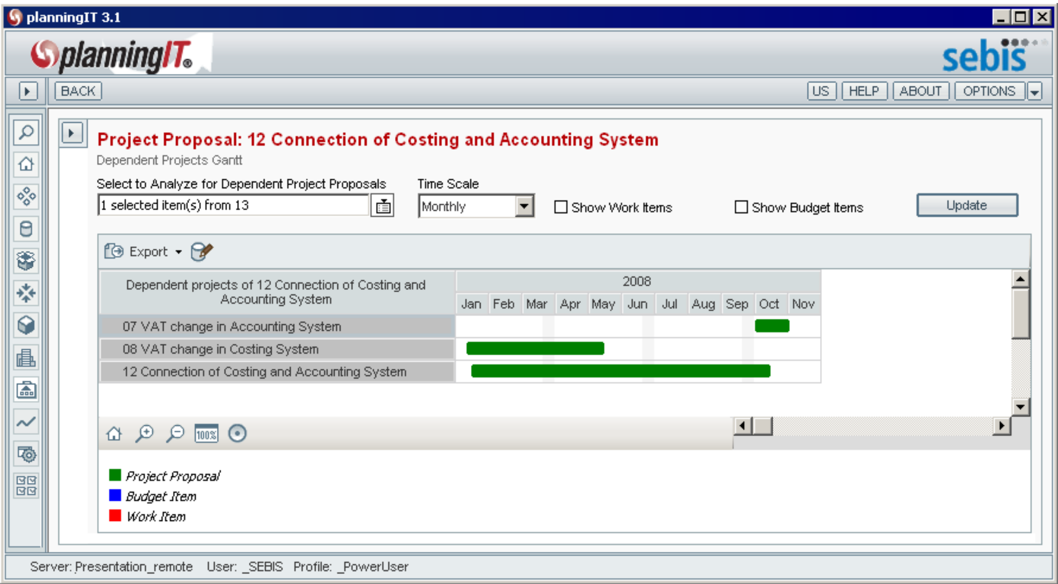


Figure 6.23.: *planningIT*- axis Synchronization Management: Gantt diagram showing dependent projects and their timelines

The procedure consistency is mostly given. It would be complete, if project delays could be entered, or the changing of planned end dates of projects caused an impact analysis concerning potential negative influences on other projects.

[8] According to alfabet, *planningIT*'s methodology does not directly cover the project execution management, which is addressed using integration to project portfolio management solutions, such as e.g. HP's BTO or CA's Clarity. Nevertheless, the upcoming version of *planningIT* will support milestone planning including a tracking process.

[9] According to alfabet, analysis on the level of "all projects" has turned out to be impractical given hundreds of concurrent projects with typically 10 or 20 affected architecture elements per project.

The procedure integration is mostly given. Besides, the missing functionality, elements entered in preceding scenarios, e. g. demands entered in the *Demand* module as well as the project proposals maintained within the *portfolio* module, can be used to support the basic functionalities of synchronization management.

Rating: 5

6.2.5. Strategies and Goals Management

planningIT provides the *Value* module to manage, analyze, and monitor strategies and goals using the concept of *value nodes*[10]. Thereby, *planningIT* differentiates between *strategic goals* and *organizational goals*[11]. In the following scenario simulation only strategic goals are used. Goals can be structured hierarchically to decompose a defined strategy to real goals operationalizing the strategy. Figure 6.24 visualizes two strategies and their decomposition into goals.

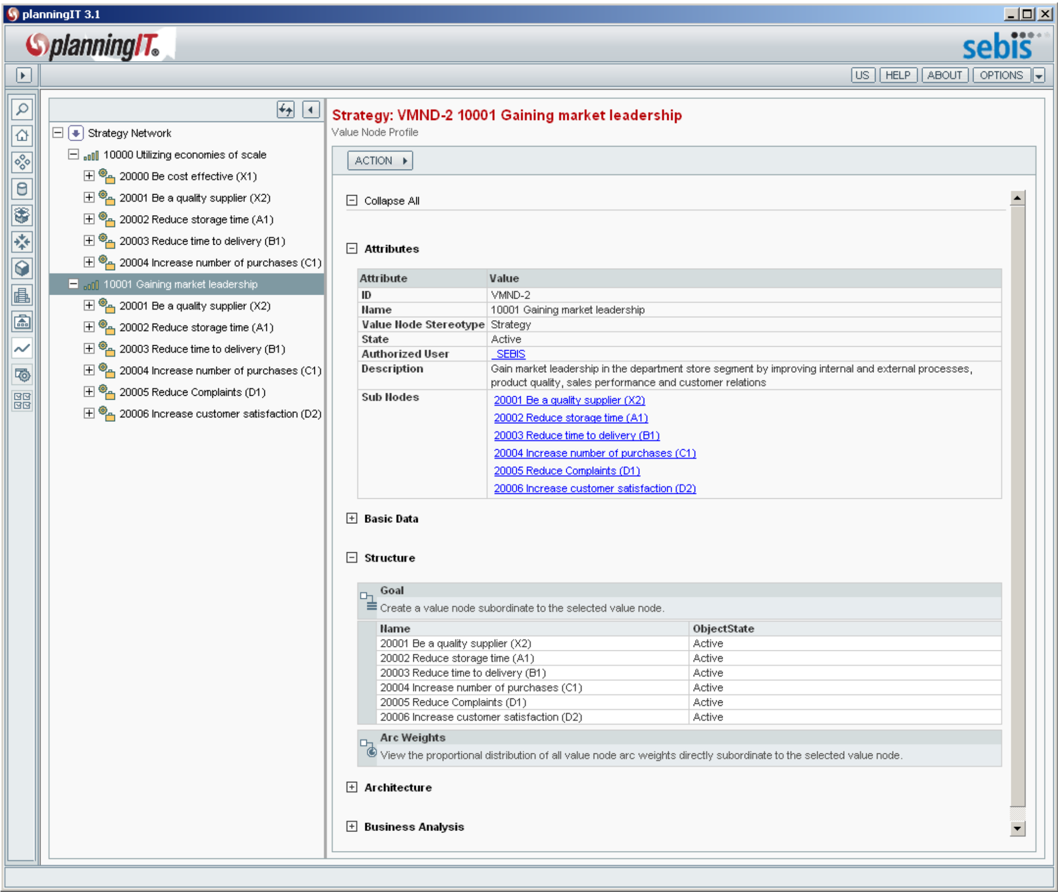


Figure 6.24.: *planningIT*- axis Strategies and Goals Management: Value management illustrating strategies and goals

[10]Value nodes support the creation of hierarchically structured configurations of goals and strategies using evaluation types and indicators to measure their achievement.

[11]Whereas, a strategic goal is a goal defined to support a strategy, which is specified for an organizational unit, an organizational goal is a goal directly defined for an organizational unit.

For every goal relating to a strategy a weighting is defined. By default the weighting for every goal is equal. The weighting can be edited through a series of pairwise comparisons. The resulting weighting can be graphically displayed as visualized in Figure 6.25. Using the built-in query functionality of *planningIT*, a report displaying the achievement of goals can easily be created, a possible result is visualized in Figure 6.26. For evaluating the achievement of each goal a set of metrics (called *indicators* within *planningIT*) can be defined. Thereby, these indicators are specified by a query or calculation in the repository.

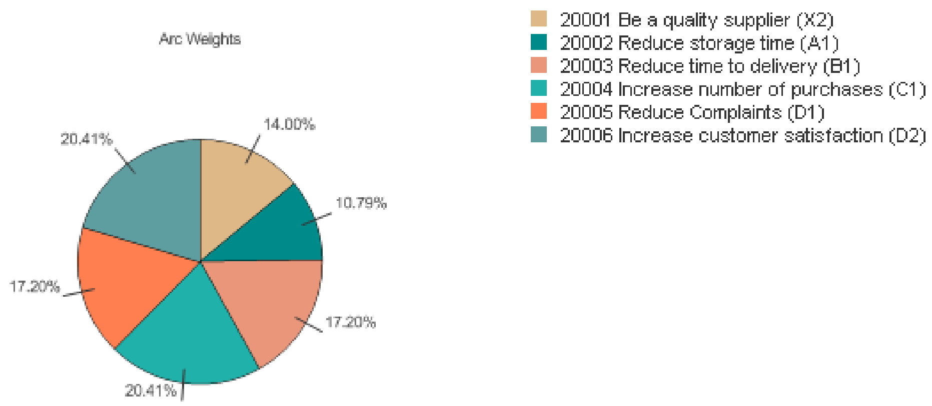


Figure 6.25.: *planningIT*- axis Strategies and Goals Management: Pie chart showing weights of goals

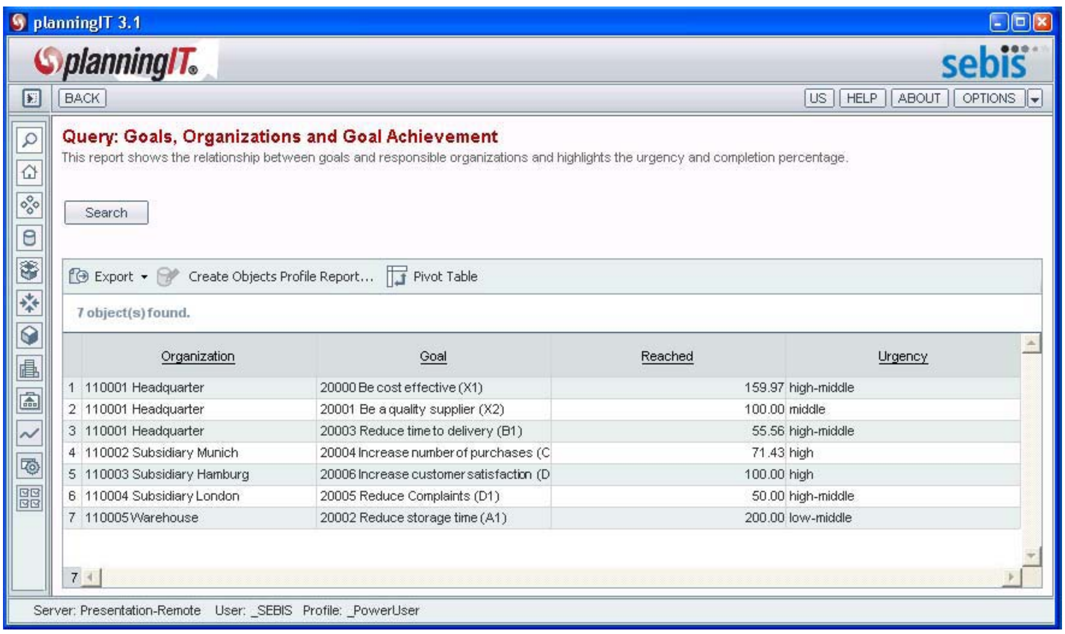


Figure 6.26.: *planningIT*- axis Strategies and Goals Management:

For every goal, elements can be defined that are related to the selected goal. For instance, relevant demands and project proposals that are aimed at implementing the goal can be browsed as well as affected architecture elements can be defined and viewed. In the same way, *planningIT* provides the possibility to navigate from the demands or project proposals to the goals, they are (partly) implementing. Custom Reports displaying the dependencies between strategies, goals, demands, projects, and application systems could be easily created, however the generation of a graphical representation as requested in Section 4.2.5 (cf. Figure 4.33 or Figure 4.35) is not possible.

The procedure consistency is completely given. Strategies, goals, and metrics to measure the achievement of goals can be defined and data can be entered using the built-in functionalities of *planningIT*. Some of the objectives (cf. Figure 4.33 or Figure 4.35) are not available in *planningIT* but custom reports can be created providing similar information.

The procedure integration is completely given, strategies can be specified and mapped to goals. Goals can be entered and mapped to existing demands or projects.

Rating: 7

6.2.6. Business Object Management

Using the *Inventory* module of *planningIT*, it is possible to define information flows between application systems and the operations performed on business objects. Thereby, *planningIT* distinguishes between *Business Objects* representing entities relevant to the enterprise's business domain and concrete, logical instances thereof, which are called *Business Data*. Business objects can therefore be exchanged between business processes or business functions, whereas business data are processed in or exchanged between application systems. The type of an information flow as well as the operations performed on business objects/data (create, read, update, or delete) can be entered.

Analyzing the information flow between application systems can be performed in two distinct ways. On the one hand, the user can start with the application system and scrutinize the information architecture, which visualizes the business data the selected object operates on as well as the Information flows, which are associated with the selected object. On the other hand, the user can view the business data profile of a selected business data object, which lists the application systems performing operations on it. Graphical analysis are additionally supported as e. g. illustrating the inter organizational information flow between application systems spread over different organizational units (cf. Figure 6.27).

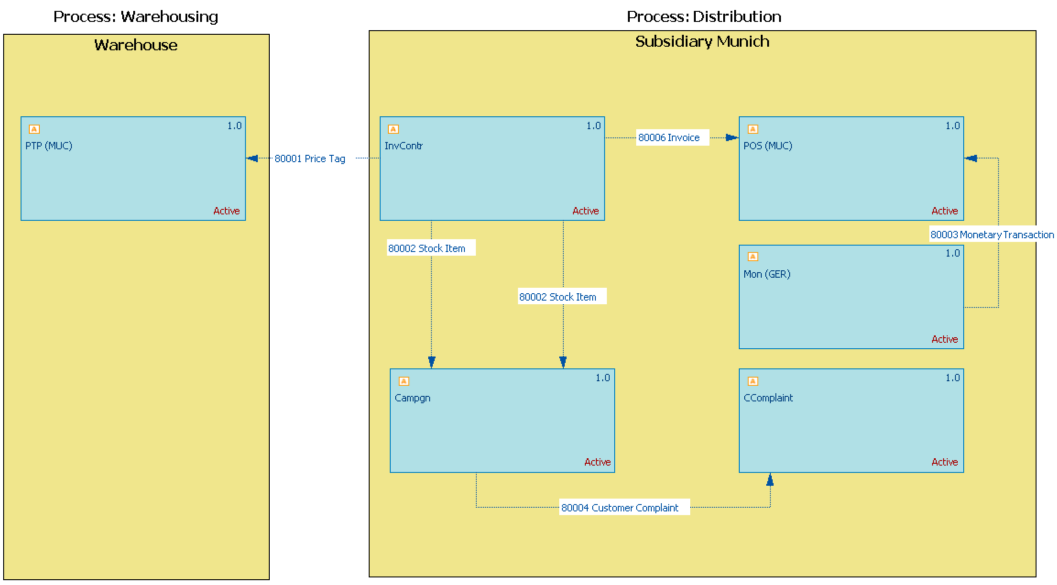


Figure 6.27.: *planningIT*- axis Business Object Management:
Cluster map visualizing inter organizational information flows

Concerning the operations performed on the data objects, *planningIT* provides further graphical analysis possibilities, as e. g. the illustration of CRUD information on graphlayout diagrams (cf. Figure 6.28). Additionally, it is also possible to create a CRUD Matrix showing the application systems, the business objects, and the operations performed on them (for an example see Section 6.2.7 Figure 6.29).

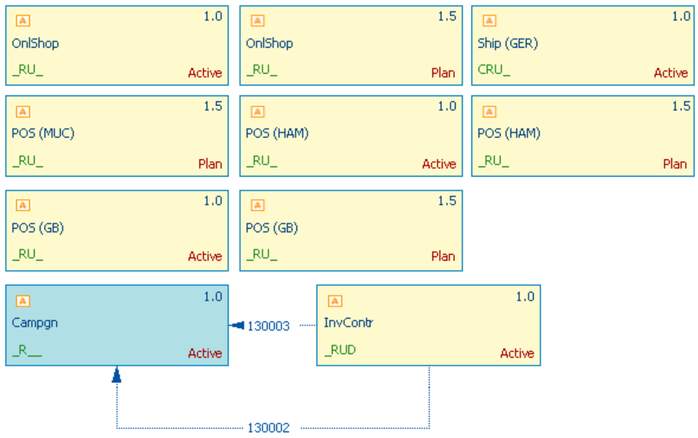


Figure 6.28.: *planningIT*- axis Business Object Management: Graphlayout map displaying CRUD information for the business object "Stock Item" (highlighting potential data inconsistencies via colorcoding)

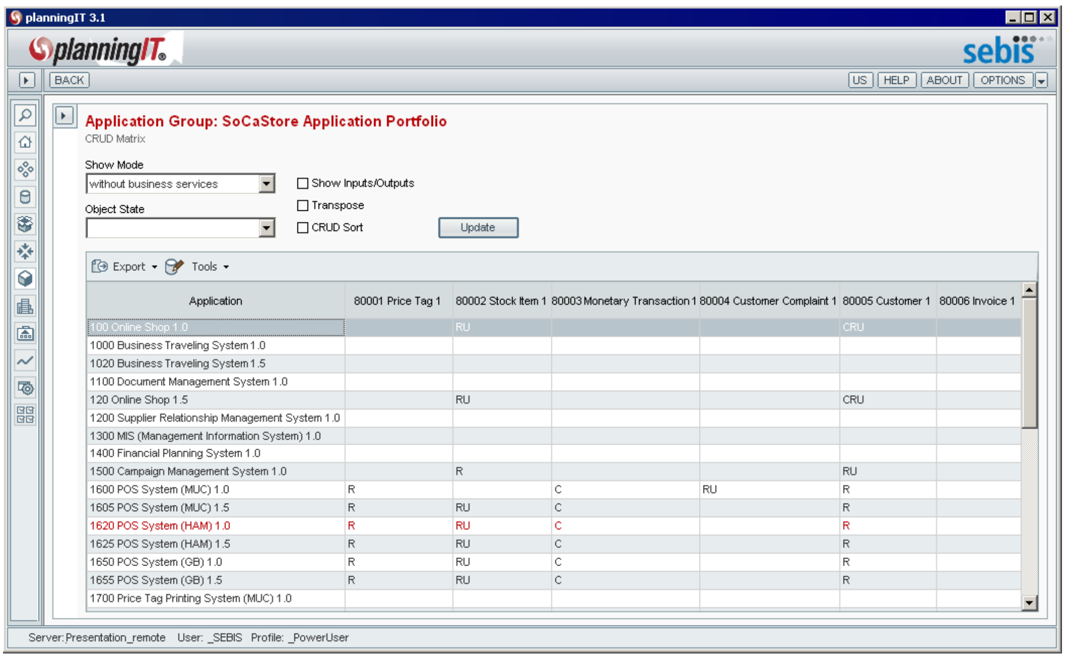


Figure 6.29.: *planningIT*- axis Business Object Management: CRUD Matrix showing application systems and business objects with the operations performed on them

The procedure consistency is completely given, the concepts provided by *planningIT* bear strong resemblance to the concepts of the scenario.

The procedure integration is completely given. Modeling the business objects and the information flows between application systems can be performed using the previously defined artifacts as e. g. application systems, or organizational units from preceding scenarios.

Rating: 7

6.2.7. SOA Transformation

Services can be entered and used within *planningIT* conforming to the built-in methodology. Thereby, the concept of service as introduced in the scenario description is mapped to *Business Functions* in *planningIT*. A business function in *planningIT* defines the functionality, which an application system provides reacting on a business service request from a business process. For each of these functionalities, application systems can be defined, which offer business services realizing the business function. If service level agreements should be defined for a service, indicators can be used to allow analyses. An overview about a business function and its related elements is visualized in Figure 6.30.

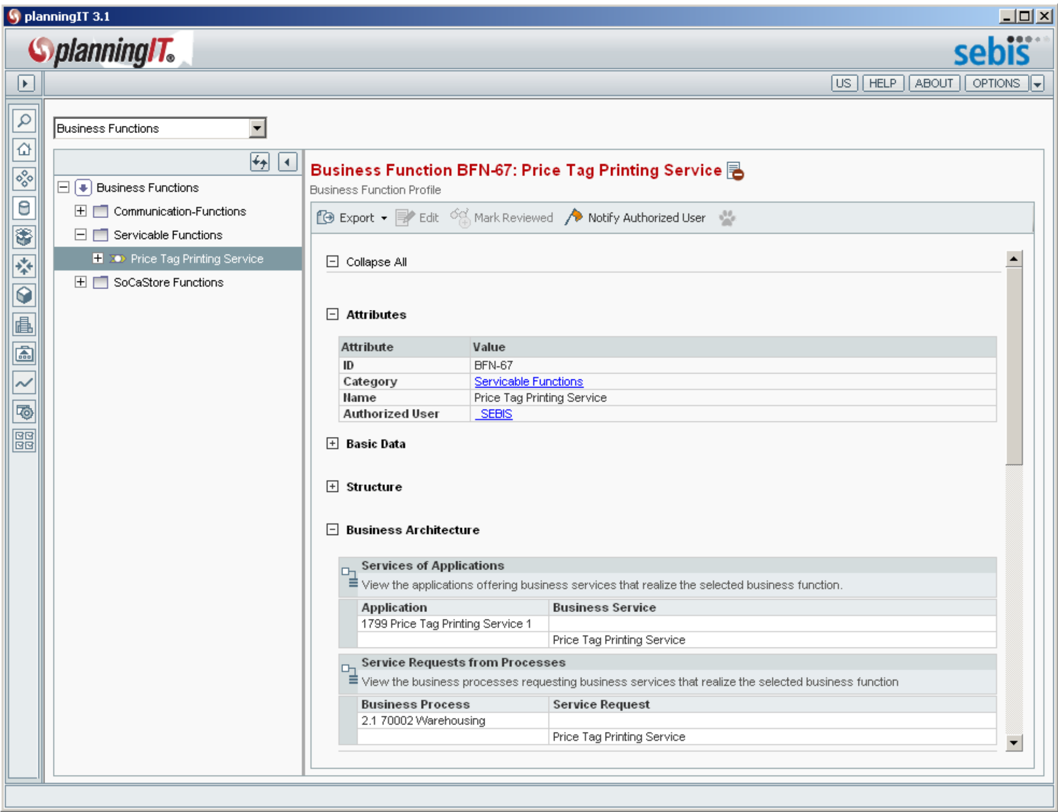


Figure 6.30.: *planningIT*- axis SOA Transformation: Business function overview with its related elements

Providing support for the top-down analysis, searching for possible candidates for services, the custom report functionalities of *planningIT* can be used to define a custom report e. g. listing application systems and the number of different organizational units, where the system is used (see Figure 6.31). Further, a similar report can be created listing the application systems, the business process they support as well as the criticality of the business process. Additionally, graphical analyses are supported by *planningIT* as e. g. the information contained within the two previously described reports can be combined to a portfolio matrix illustrating the criticality of the supported business process on the x-axis, and the number of using organizational units on the y-axis.

	Name of Application	Number of organizational units
1	100 Online Shop	1
2	1500 Campaign Management System	4
3	200 Inventory Control System	20
4	300 Monetary Transactions System (GER)	10
5	350 Monetary Transactions System (GB)	3
6	400 Product Shipment System (GER)	1
7	510 Accounting System	1
8	600 Costing System	1

Figure 6.31.: *planningIT*- axis SOA Transformation: Report showing the applications and the number of organizational units they are used in

Likewise, a custom report can be used to support the bottom-up analysis. Figure 6.32 visualizes a custom report listing the organizational units, the application systems, the business objects, and the operations performed on them. Similar reports could be easily created listing application systems that e. g. will be changed in the near future.

	Organization	Application	Business Data	Business Function	Read	Update	Delete	Create
1	110001 Headquarter	100 Online Shop	80002 Stock Item	90007	x	x		
2	110001 Headquarter	100 Online Shop	80002 Stock Item	90008	x	x		
3	110001 Headquarter	100 Online Shop	80005 Customer	90007	x	x		x
4	110001 Headquarter	100 Online Shop	80005 Customer	90008	x	x		x
5	110001 Headquarter	1500 Campaign Management System	80002 Stock Item	90003	x			
6	110001 Headquarter	1500 Campaign Management System	80002 Stock Item	90004	x			
7	110001 Headquarter	1500 Campaign Management System	80005 Customer	90003	x	x		
8	110001 Headquarter	1500 Campaign Management System	80005 Customer	90004	x	x		
9	110001 Headquarter	200 Inventory Control System	80002 Stock Item	90001	x	x	x	

Figure 6.32.: *planningIT*- axis SOA Transformation: Custom report showing organizational units, application systems, business objects, and the operations performed on them

Concerning the transformation process, *planningIT* provides the possibility to use process support maps or time interval maps illustrating the performed changes (cf. Figure 6.33).

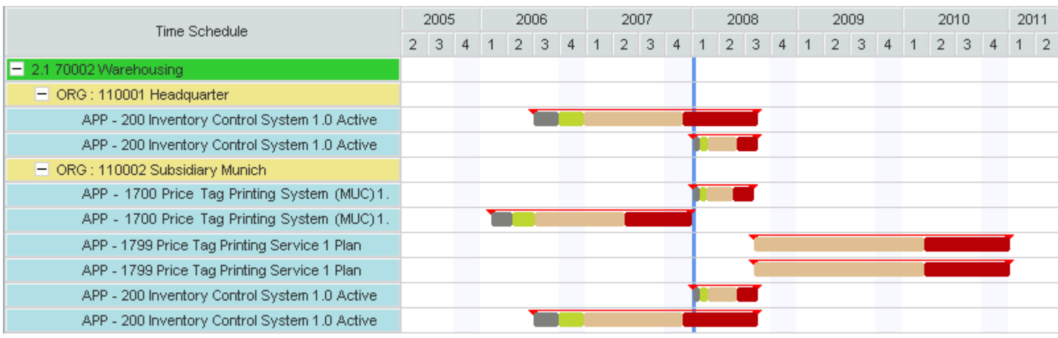


Figure 6.33.: *planningIT*- axis SOA Transformation: Time interval map visualizing changes of the application landscape

The procedure consistency is mostly given, services, service level agreements and their relations to other elements can be entered and analyzed using standard as well as custom reports as described above.

The procedure integration is completely given, elements entered in preceding scenarios, can be used.

Rating: 6

6.2.8. IT Architecture Management

The *IT Standards* module of *planningIT* provides functionalities to define IT standards as blueprint elements, solution elements, architectural blueprints, and architectural solutions like required for the simulation of the scenario IT architecture management (see Section 4.2.8). Thereby, blueprint elements (mapped to *component category* objects within *planningIT*) can be used to classify and group solution elements (mapped to *components*) as visualized in Figure 6.34. An architectural blueprint (mapped to a *master platform* in *planningIT*) specifies the architecture consisting of tiers or layers. Subsequently, an architectural solution (called *standard platform* in *planningIT*) realizes a specified architectural solution utilizing solution elements.

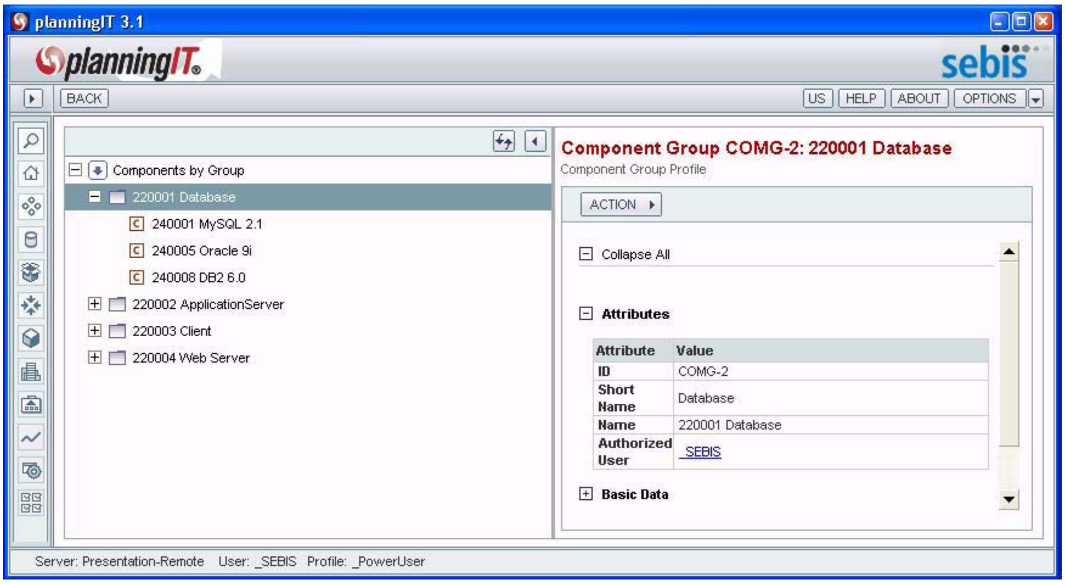


Figure 6.34.: *planningIT*- axis IT Architecture Management: Mapping of blueprint elements to component categories

A standard report as the one visualized in Figure 6.35, which lists all applications using a specified architectural solution, is available. Furthermore, graphical analyses are supported using cluster maps visualizing, e. g. the conformance with architectural solutions (cf. Figure 6.36). Additional analysis concerning the whole application landscape can be performed by using custom reports build with the query builder of *alfabet eXpand*. Figure 6.37 lists all application systems as well as the architectural blueprint and solution they use. The possibility to represent the results of such queries as e. g. bar charts is currently not provided.

The procedure consistency is completely given. The concepts of architectural blueprints, architectural solutions, blueprint elements, and solution elements can be easily mapped to the corresponding classes in the information model of *planningIT*.

The procedure integration is completely given. Objects defined in other scenarios, e. g. application systems and organizational units can be used and reports as well as graphical representations can be created using the built-in functionalities or by defining a custom report.

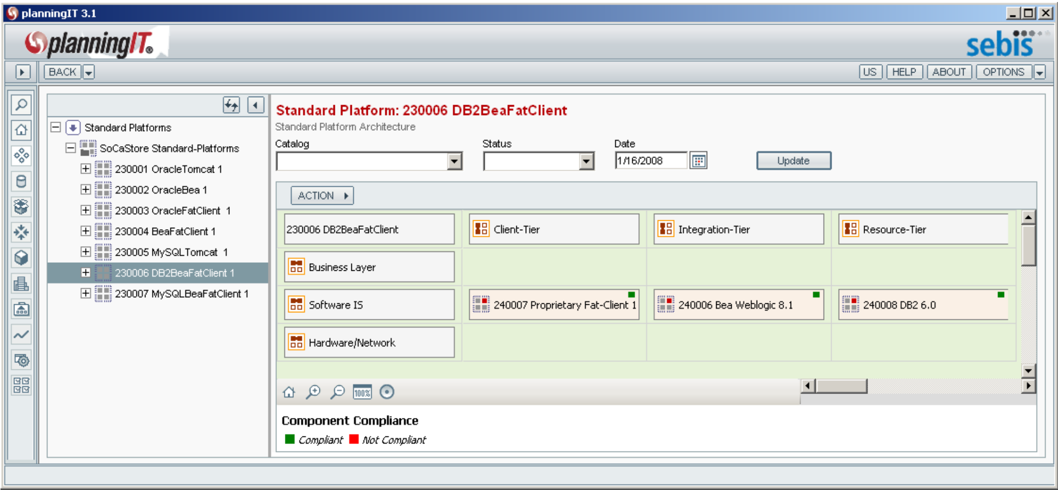


Figure 6.35.: *planningIT*- axis IT Architecture Management: List of application systems using the given standard platform highlighting compliance to the technology roadmap

Figure 6.36.: *planningIT*- axis IT Architecture Management: Cluster map showing conformance to an architectural solution

Rating: 7

Master_Platform	Standard_Platform	Application
⊟ 210003 4-Tier-Thin-Client-Architecture	⊟ 230001 OracleTomcat	100 Online Shop
		120 Online Shop
		1200 Supplier Relationship Management System
		1500 Campaign Management System
		1900 Customer Complaint System
		2000 Customer Satisfaction Analysis System
		2020 Customer Satisfaction Analysis System
		2300 Accounting and Costing System
		2400 Knowledge Managment System
		500 Accounting System
	⊟ 230002 OracleBea	700 Human Resources System
		800 Data Warehouse
	⊟ 230005 MySQLTomcat	1300 MIS (Management Information System)
		1320 MIS (Management Information System)
		1340 MIS (Management Information System)
		400 Product Shipment System (GER)
		900 Fleet Management System

Figure 6.37.: *planningIT*- axis IT Architecture Management: List of all application systems as well as their standard and master platforms

6.2.9. Infrastructure Management

Using the menu entry *Technical Architecture* of the *Inventory* module of *planningIT*, information can be entered and accessed for managing the enterprise's infrastructure. Thereby, the concept of solution elements (mapped to *components* in *planningIT*) is of vital importance. Solution elements can be classified and grouped to blueprint elements (called *component category* within *planningIT*), e. g. databases, which include all different databases that are in use (cf. Figure 6.34).

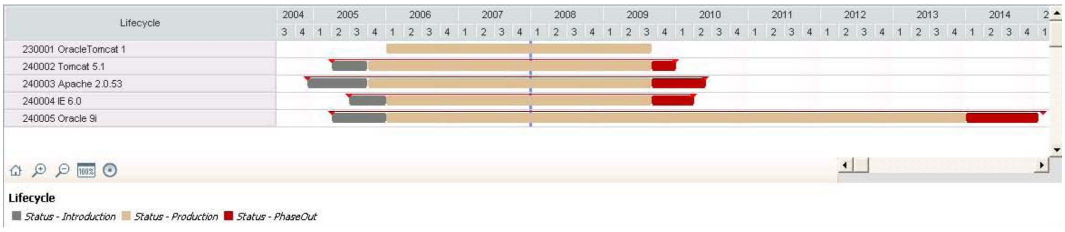


Figure 6.38.: *planningIT*- axis Infrastructure Management: Databases and their lifecycle information

For every solution element information is maintained, e. g. the deployment at a specific organizational unit or location, or the lifecycle information of the solution element including the date for end of support. Thus, graphical representations, which support analyses of solution elements running out of support, can automatically be generated as shown in Figure 6.38. Similar analyses can be performed using a custom report listing the solution elements, their end of support, and the current status (see Figure 6.39). Financial aspects, as e. g. license and operating cost can be entered using *evaluations*[12].

Supporting the analysis of the application landscape in respect to infrastructure issues, *planningIT* provides capabilities to analyze the current landscape using graphical representations. An example is shown in Figure 6.40, which visualizes for every application system, the used database, the location

[12]See Section 6.2.5 for a detailed description.

		Status					
		Introduction		PhaseOut		Production	
Component	Support_End_Date	Start_Date	End_Date	Start_Date	End_Date	Start_Date	End_Date
240001 MySQL	12/31/2010	1/1/2005	9/30/2005	6/16/2009	6/1/2010	10/1/2005	6/15/2009
240005 Oracle	1/31/2015	4/1/2005	12/31/2005	1/1/2014	12/31/2014	1/1/2006	12/31/2013
240008 DB2	6/30/2011	6/1/2004	5/31/2005	1/1/2010	12/31/2010	6/1/2005	12/31/2009

Figure 6.39.: *planningIT*- axis Infrastructure Management: Report showing solution elements with their end of support date as well as with additional status information

the application system is deployed, and the status of the component. Using this visualization, a new demand for replacing the database system ”240001 MySQL”, which is in danger of running out of support, is created that results in a project proposal.

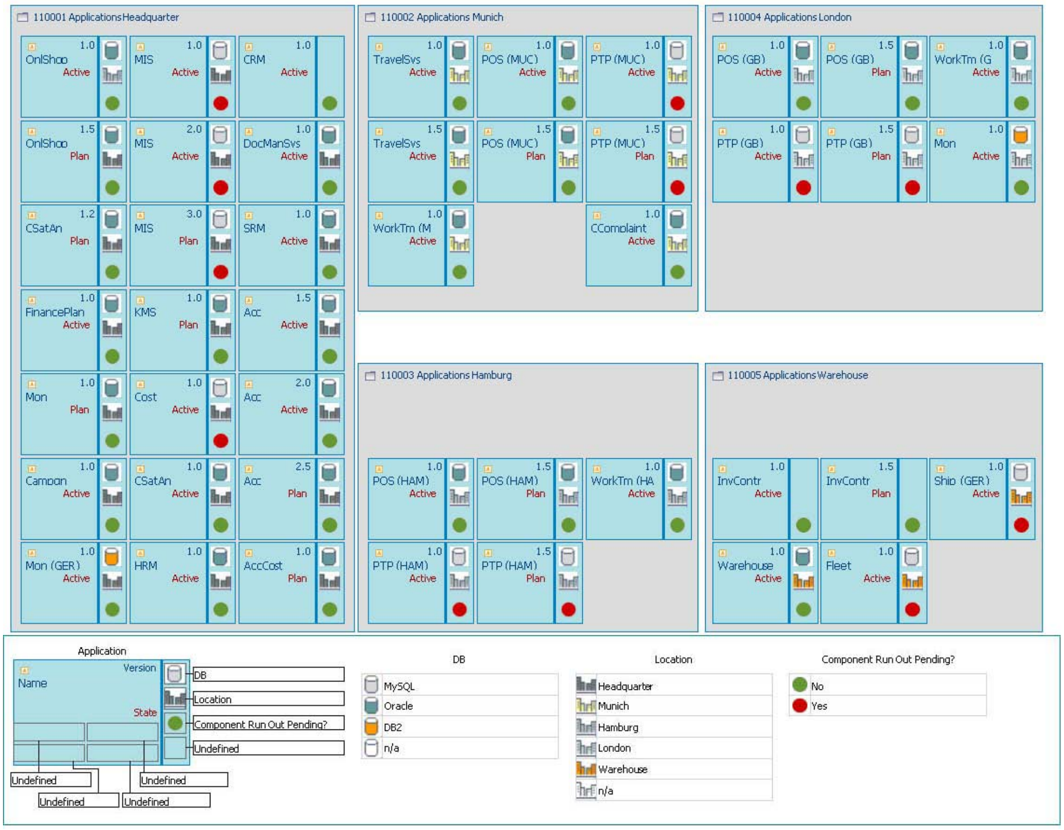


Figure 6.40.: *planningIT*- axis Infrastructure Management: Cluster map visualizing the used database, location of the application, and status of the component

Using the query functionalities provided by *planningIT*, the report visualized in Figure 6.41 can be created, which lists all application systems affected by the consolidation.

The procedure consistency is mostly given. Solution elements can be entered and their deployment can be defined. Analysis supporting the management can be conducted using *planningIT*'s standard reports complemented by a few queries and custom reports. Demands and project proposals cover the evolution aspect of the scenario.

The procedure integration is completely given, information entered in preceding scenarios can be reused.

	Application	Component	Device	Location	Support End Date
1	1300 MIS (Management Information System)	240001 MySQL	Server Headquarter	120001 Headquarter	12/31/2010
2	1320 MIS (Management Information System)	240001 MySQL	Server Headquarter	120001 Headquarter	12/31/2010
3	1340 MIS (Management Information System)	240001 MySQL	Server Headquarter	120001 Headquarter	12/31/2010
4	1700 Price Tag Printing System (MUC)	240001 MySQL	Server Munich	120002 Munich	12/31/2010
5	1705 Price Tag Printing System (MUC)	240001 MySQL	Server Munich	120002 Munich	12/31/2010
6	1720 Price Tag Printing System (HAM)	240001 MySQL	Server Hamburg	120003 Hamburg	12/31/2010
7	1725 Price Tag Printing System (HAM)	240001 MySQL	Server Hamburg	120003 Hamburg	12/31/2010
8	1750 Price Tag Printing System (GB)	240001 MySQL	Server London	120004 London	12/31/2010
9	1755 Price Tag Printing System (GB)	240001 MySQL	Server London	120004 London	12/31/2010
10	600 Costing System	240001 MySQL	Server Headquarter	120001 Headquarter	12/31/2010
11	900 Fleet Management System	240001 MySQL	Server Warehouse	120005 Warehouse	12/31/2010

Figure 6.41.: *planningIT*- axis Infrastructure Management: Report showing all applications, which are affected by the database consolidation

Rating: 6

6.3. Tool Vendor's Profile

planningIT: Managing the Business of IT
Paula Ziehr, Marketing, alfabet AG

planningIT enables companies to strategically plan their IT based on information from the enterprise architecture and in alignment with the business. It is unique in tightly coupling business priorities and IT returns with current and future initiatives. Competitive point solutions offer modeling or repository functions with a static view of priorities, architecture and infrastructure. alfabet provides a holistic, integrated and collaborative approach that offers continuous transparency into how IT and business information, processes and roles are changing and need to be managed over time.

Managing the Business of IT

planningIT supports organizations in managing the business of IT by providing the means to:

- target investment in areas that advance the enterprise strategically
- identify weaknesses that hinder achievement of enterprise goals
- determine how to improve on identified weaknesses
- ensure success of planned improvement initiatives

Against the backdrop of corporate governance, *planningIT* delivers information to management on corporate performance to better understand how effectively and efficiently IT is supporting business goals and where processes can be improved. It lets an organization define roles, responsibilities and processes within IT and between IT and business to ensure every stakeholder's participation in the strategic IT planning process. This promotes business/IT alignment and accountability for architectural issues and domains. Supporting the corporate and IT planning activities at all levels of management, *planningIT* ensures integration of the strategic and operational layers of management.

planningIT enables IT/Business Alignment

Three pre-requisites need to be met in order to plan and manage an organization's IT so that it can optimally support the business: access to all architecturally-relevant data, well-defined processes to avoid redundant efforts and ensure seamless interaction between users involved in the process, and involvement of and communication between all relevant stakeholders. *planningIT* supports an organization in meeting these requirements.

planningIT helps overcome the challenge of architecture information that is widely dispersed throughout the organization by providing a single, structured and organized platform of enterprise architecture information describing the business/IT relationship. The Logical Inventory is the information source for all stakeholders. Workflows, monitors and ownership accountability ensure data is kept up-to-date.

The product's process platform bundles best practice activities into functional modules addressing the individual needs of the various stakeholders involved in the process of IT planning. Modules include Business Demand Management, Application Architecture Management, Program Portfolio Management, Enterprise Architecture Management, Value Management, Enterprise Strategy and Master Planning, Release Management and Compliance Management.

planningIT captures all aspects of the architecture and can inform on different views onto the architecture making the product appealing to a wide range of users and thus promoting a collaborative IT planning process in the organization. Its ability to aggregate information at various levels and visualize these in reports, graphs and charts makes it the optimal platform for business/IT communication.

Getting Business Value out of IT: Managing Opportunities, Risks and the Enterprise Architecture with planningIT

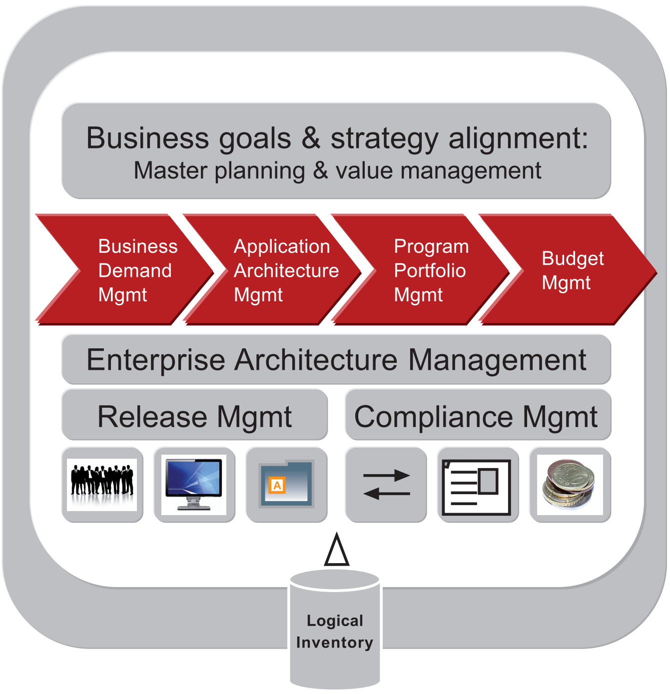
While it is clear to most that enterprise architecture is an integral part of corporate planning, it is essential to understand that only a data-driven, process-oriented approach can support this effectively.

Changes in corporate strategy and technology result in new demands on the architecture despite best-effort planning. To manage this volatility, steady and continuous process and information management are required - adapting and always moving towards the target architecture. Only an inventory that is fed by transactions resulting from the planning process can ensure decisions and plans are based on up-to-date and valid information for the current state of the architecture as well as all future states. It supports the organization in taking advantage of opportunities that arise and creating synergy among parallel transformational projects.

Inadequate technical assessment and business cases lead to the failure of over 50% of all IT projects. Knowing this and the fact that 50% of the functionality delivered by IT is not even used, IT organizations need to put more effort into architectural governance - architectural governance such as delivered by *planningIT* in the form of:

- scenario and impact analysis to identify architectural conflicts
- solution design including cost and risk analysis
- multiple and multi-dimensional portfolio analysis
- definition of standards platforms and blueprints
- checking for compliance to IT standards and granting of waivers
- creation of master plans and roadmaps based on the inventory

Summarizing, *planningIT* helps organizations to mature in their ability to manage the business of IT and deliver business value by providing a data-driven, process-oriented and collaborative platform for pro-active IT planning and management.



CHAPTER 7

BOC (ADOit)

Contents

The concept of *BOC* is to deliver the meta plattform *ADOit* which offers standard functionality that usually is further adapted regarding the specific requirements of customers. Out of the box, *ADOit* is delivered with a default configuration containing a standard information model and standard functionality, like a view generator and a query language. For this survey *BOC* treated the objectives like a customer project. Thus, for matching further requirements of the *SoCaStore* *BOC* consultants configured a specific *ADOit* configuration for the survey using the customization capabilities of *ADOit*. This version is referred to as *ADOit for SoCaStore* in the evaluation. However, the *out of the box* version was evaluated as well and is referred to as *ADOit standard configuration*.

ADOit consists of the following applications:

The *ADOit IT Architecture and Service Management Toolkit* is the client for creating and maintaining data as well as for creating visualizations and reports.

The *ADOit administration toolkit* is the administration tool providing e. g. user management and component management.

Homer is a component, which uses the tabular and function-orientated features of Microsoft Excel, in order to collect and administrate information conveniently.

ADOit web client is the lightweight access component.

7.1. Evaluation of Specific Functionality

This chapter outlines the detailed results of simulating the scenarios from Section 4.1 with *ADOit 3.0*.

7.1.1. Importing, Editing, and Validating

For the import *ADOit* provides a standard import function based on a scripting language. Therefore, structured data from different sources, like *MS Excel files*, *XML-based files*, or *ODBC sources* can be imported. For importing the *SoCaStore* data from the MS Excel spreadsheet two adaptations have to be made. Firstly, the information model, which will be shipped with the standard installation, has to be adapted in order to fit to the information model of *SoCaStore*. Secondly, the import mechanism based on a scripting language called *AdoScript* has to be adapted. The scripting language supports data transformations in the importing process, e. g. splitting comma separated values maintained in one spreadsheet column. Thus, the import can be flexibly adapted by programmers, who have experience with the language, but usually, *BOC* consultants will make the necessary adaptations. With those preparations an MS Excel spreadsheet can be imported without adaption of the spreadsheet itself and all the data of *SoCaStore* data is imported as required. Figure 7.1 shows the import process realized in *ADOit*.

Furthermore, the tool supports a consistency check of data during the import. After the import an error report is provided, e. g. showing missing attributes, or incorrect data types. It is recommended to copy the report and save it in a separate document, because it is not saved by the tool. After the import, consistency checks are further available by the *Analysis component* of *ADOit*. This functionality checks, whether all attributes of the objects of the selected models contain valid data. As standard, there is no distinction between mandatory and optional attributes and all attributes are assumed as mandatory. However, such a check can be configured in the *ADOit administration toolkit*.

ADOit enables users to edit data in a convenient way. Only changes that are compatible with the underlying information model are allowed. One way for editing data is clicking on an object in a visualization. Thus, the property box of the object, called *notebook* is opened containing several tabs, like *description*, *references*, and *time intervals*, where changes can be made. Figure 7.2 shows the

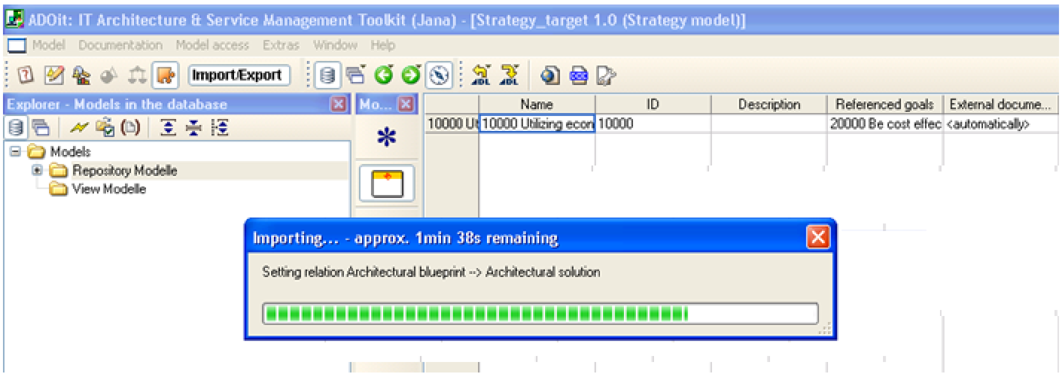


Figure 7.1.: *ADOit* - axis Importing, Editing, and Validating: The importing process

references tab, where e. g. a new relationship to another object can be created.

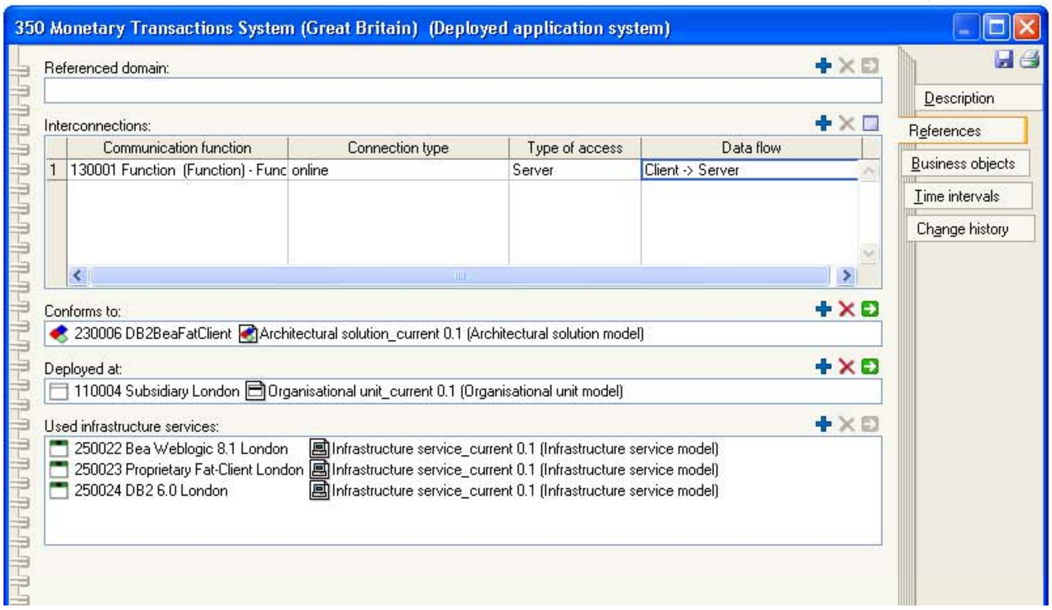


Figure 7.2.: *ADOit* - axis Importing, Editing, and Validating: Editing the relationships of a application system

Another way of editing data is in a *table view*, which shows all objects of a certain class, like the one provided in Figure 7.3 displaying all application systems with selected attributes. The *tabular view* provides functions like hiding columns and sorting columns, as well as filtering mechanisms to show or hide objects based on selected attributes. Besides, it is possible to provide picklists or to link to an external documentation, e. g. a file from the file system or an URL. Furthermore, from the *tabular view* it can be switched and switched back to a *graphic view*, showing all objects of a selected class. By clicking on a graphically represented object, the notebook of the object will be displayed.

In addition, attributes can be set globally via the so called function *Global Change*. Target objects can be selected via query utilizing predefined or ad hoc queries. Furthermore, it is possible to use the query component to search required objects. The result list is presented in a tabular form which allows for editing the attributes of the objects. If only certain attributes at the same time should be edited, the *table view*, or parts of it can be copied back to Microsoft Excel, where changes can be made, and then

	ID	Version	Stand...	Criticality	Availability	Users	Num...	Maintenance costs
100 Online Shop	100	1.0	no	1.000000	0.992000	50	3	4000.000000
120 Online Shop	120	1.5	no	1.000000	0.995000	50	0	0.000000
200 Inventory Control System	200	1.0	no	2.000000	0.990000	30	1	2322.000000
220 Inventory Control System	220	1.5	no	2.000000	0.991000	30	10	0.000000
300 Monetary Transactions System (Germany)	300	1.0	no	2.000000	0.980000	10	3	10032.000000
350 Monetary Transactions System (Great Britain)	350	1.0	no	2.000000	0.981000	10	0	1344.000000
400 Product Shipment System (Germany)	400	1.0	no	3.000000	0.977000	15	3	3131.000000
500 Accounting System	500	1.5	no	2.000000	0.999000	10	0	4552.000000
510 Accounting System	510	2.0	no	2.000000	0.990000	10	1	0.000000
520 Accounting System	520	2.5	no	2.000000	0.999900	10	10	0.000000
600 Costing System	600	1.0	no	4.000000	1.000000	16	3	0.000000

Figure 7.3.: *ADOit* - axis Importing, Editing, and Validating: Tabular view showing application systems with selected attributes

copied back to *ADOit*. This is a very convenient way, leveraging the full editing features provided by Microsoft Excel, however the user has to take care and attention in using this feature to avoid errors arising from copy and paste.

Rating: 5

7.1.2. Creating Visualizations

ADOit provides a comprehensive approach for creating visualizations of the application landscape. All required software maps, such as cluster map (Figure 7.9), process support map (Figure 7.4), time interval map (Figure 7.5), graphlayout map (Figure 7.19), swimlane diagram (Figure 7.4), and portfolio matrix (Figure 7.6) are supported utilizing the configuration and customization features of *ADOit*.

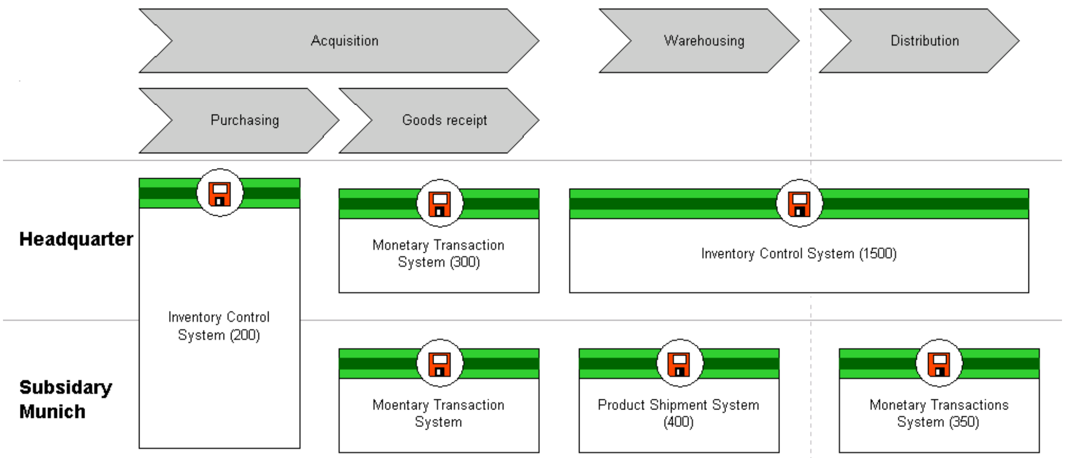


Figure 7.4.: *ADOit* - axis Creating Visualizations: Manually created process support map

Each visualization can be created manually with the *modeling* component. For each map type certain object types and drawing elements are available, that can be put per mouse-click on the canvas. Which object types are supported by which visualization can be customized. Furthermore, helpful functions like *zoom factor*, *snap to grid*, and *page layout* are provided.

On the one hand, it is possible to create visualizations, that exist detached from the rules of the information model and the objects stored in the repository. Therefore, the user is allowed to draw pictures, that e. g. contain new objects that are not available in the current object list, or to create relationships with the *standard connector* allowing relationships that are not provided by the information model.

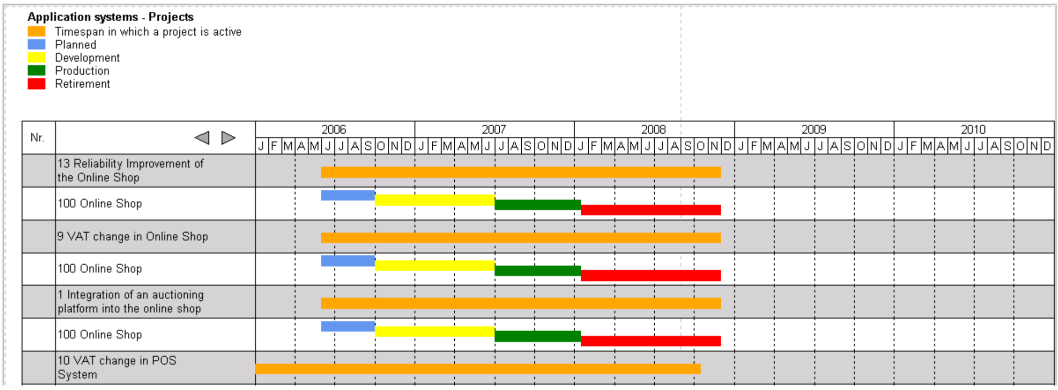


Figure 7.5.: *ADOit* - axis Creating Visualizations: Automatically created time interval map

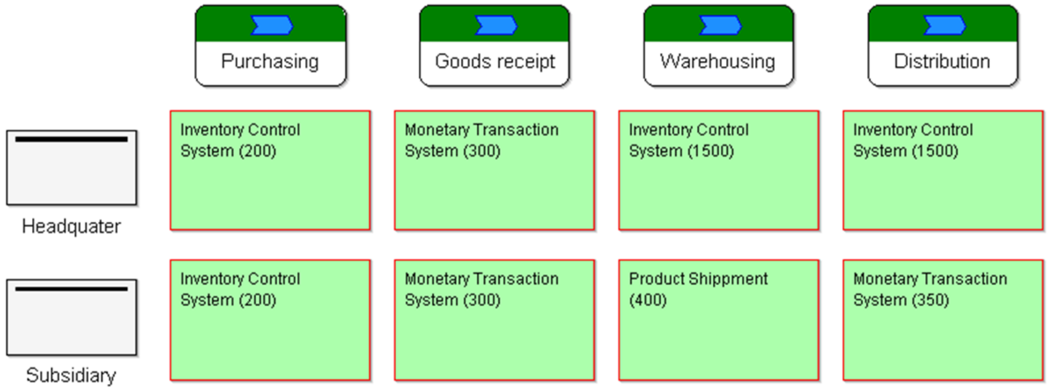


Figure 7.6.: *ADOit* - axis Creating Visualizations: Matrix showing application systems regarding their supported processes and organizational units

On the other hand, it is possible to create visualizations, that are connected to the objects in the repository and follow the rules of the information model. Usually, after creating and positioning of an object on the canvas, it can be assigned to an existing object in the repository. Due to performance reasons, objects from the repository are loaded on demand the moment they are required. Furthermore, *ADOit standard configuration* provides a cardinality check, that checks the validity of the visualization, on demand. However, *ADOit* can be customized, so that the check of validity regarding the information model occurs automatized during the modeling of a visualization.

Several predefined automatically generated software maps are shipped with the *ADOit standard configuration*, as for example portfolios, matrices, swimlane maps and spider diagrams. These visualizations are quite flexible, e. g. first selecting the requested class and then selecting all relevant objects. The automatic generation of all map types can be realized through customization, like done for *ADOit for SoCaStore*. This can be achieved by *BOC* consultants or a trained and skilled user, utilizing the *ADOit administration toolkit*, which provides basic view generation functionalities.

Within *ADOit for SoCaStore* visualizations are treated like *snapshots* that represent the application landscape at a certain point of time. Because the visualizations are not connected with the repository, changes like moving an object to another organizational unit, or resizing an object, can be performed. These changes are not transferred to the repository. However, this mechanism could be implemented via the scripting language *AdoScript* if necessary.

If changes in the repository occur, that should be visualized, two different options exist. First, if the changes are just on attribute level, these changes can be imported to the visualization using the feature *update current attributes*. Second, if the changes are on object level, e. g. deletion of a current object, the visualization has to be created new and any manual changes are disregarded.

Rating: 4

7.1.3. Interacting with, Editing of, and Annotating Visualizations

ADOit standard configuration provides different ways to interact with a visualization. One is the integrated search functionality, that searches either in the currently selected model or in all opened models. After the search a result list is shown in the lower window and by clicking on one or more results the elements will be highlighted within the selected visualization, like shown in Figure 7.7. Furthermore, several predefined searches are available, like e. g. *show all application systems connected via a certain interface*.

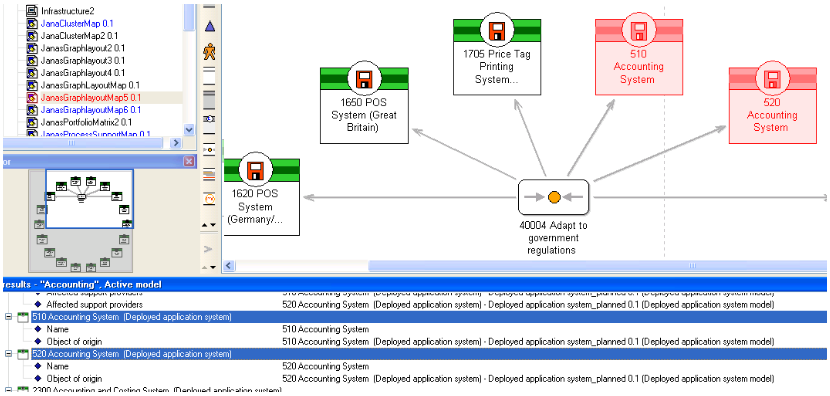


Figure 7.7.: *ADOit* - axis Interacting with, Editing of, and Annotating Visualizations: Searching objects

Besides the search capabilities, filtering mechanisms can be provided, usable for hiding objects that match a certain search criteria. Furthermore, *ADOit standard configuration* provides the capability to close symbols in order to hide a complex inner structure by the *shrink/expand* mechanism. This mechanism is e. g. used for expanding the ICT infrastructure of a certain subsidiary or to expand the software architecture of an application system.

Furthermore, navigation mechanisms regarding the visualizations are provided. For example, one way that is helpful for navigating through large scale visualizations is a helicopter window on the left side, showing a *helicopter view* picture of the entire visualization highlighting the currently viewed part of the visualization, like shown in Figure 7.7. It is also possible to click on the highlighting window and move it, thereby changing the visible part of the visualization in the viewing window. Furthermore, from each object that is from the repository it can be navigated to the *tabular view*, where all objects of the same class are available with their attributes, by clicking on the name of the object. By right mouse-click or double-click on any object in a visualization the notebook can be selected. Closely related objects can be seen by selecting the *notebook* clicking the tab *references*. However, navigating from there to the closely related object in the selected visualization is not directly supported.

Another requirement is the support of manual adaptations of visualizations, like resizing of an object. Manual changes regarding visualizations that are not connected with the repository are always possible. Furthermore, multiple changes concerning more than one element of a visualization, like clicking more than one application systems and resizing it, change style, or change symbol is supported.

Automatically created visualizations within the*ADOit for SoCaStore* can be adapted as well. *ADOit for SoCaStore* does not distinguish between changes that do or do not alter the visualization's semantic, because the changes are not transferred to the repository. However, *BOC* consultants point out, that this could be configured otherwise.

Once a report is created, there exist some basic adaptation functionalities, like sorting columns regarding every attribute. Whereas, it is not possible to hide objects in a once created report, a new report can be created, selecting only the required objects.

In the *ADOit standard configuration* some annotations for visualizations for basic attributes are already available. Further annotations for basically any available attribute can be configured and customized using the customization features of *ADOit*. Therefore, the representation of annotations is very flexible, e. g. changing background colors, using small symbols like signs or traffic lights, or changing the objects size, like shown in Figure 7.8, is supported. Furthermore, it can be specified, if the annotation should change using thresholds or distinct values. Besides, it can be defined, if the annotation should work for every visualization within *ADOit* that has this annotated attribute, which is nice, or if it should work just for selected objects in a visualization. Because, each annotation can be selected from the tools menu, more than one annotation, e. g. the use of traffic lights and background color, is possible. Figure 7.8 shows application systems with several annotations, as the number of users by background color, the availability of the application system by traffic lights, and the modifying projects by their project number.

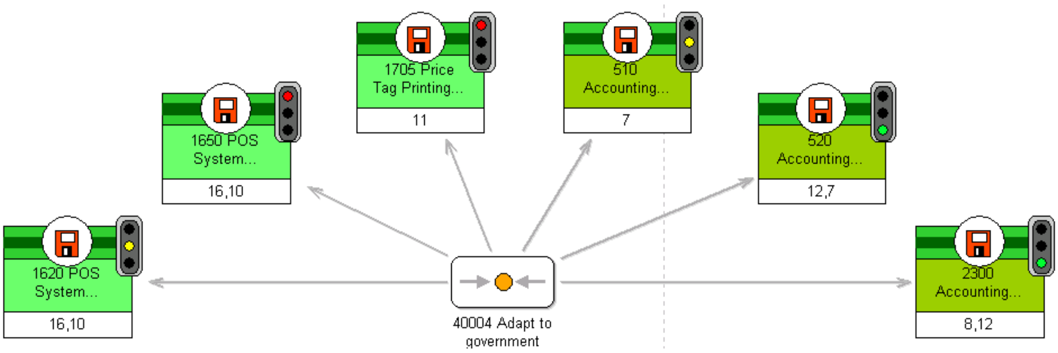


Figure 7.8.: *ADOit* - axis Interacting with, Editing of, and Annotating Visualizations: Application systems with several annotations

In *ADOit for SoCaStore* all required annotations are available, like annotating the number of users, architectural solutions conformance, availability, or maintenance costs. Figure 7.9 shows an example where the application systems are colorized based on the number of users within an automatically created visualization.

The creation or adaption of annotations can be done with the *ADOit administration toolkit*. There, a flexible scripting language called *GraphRep* can be utilized, which can be used from a skilled user after familiarization. During the creation process it is also determined, where in the tools menu the annotation function should be added.

Rating: 4

7.1.4. Communication and Collaboration Support

ADOit provides a role based access control that can handle read and write right access for user groups down to attribute level. This can be done within the *ADOit administration toolkit* component *User Management*. There, a configuration of user groups and roles is available.

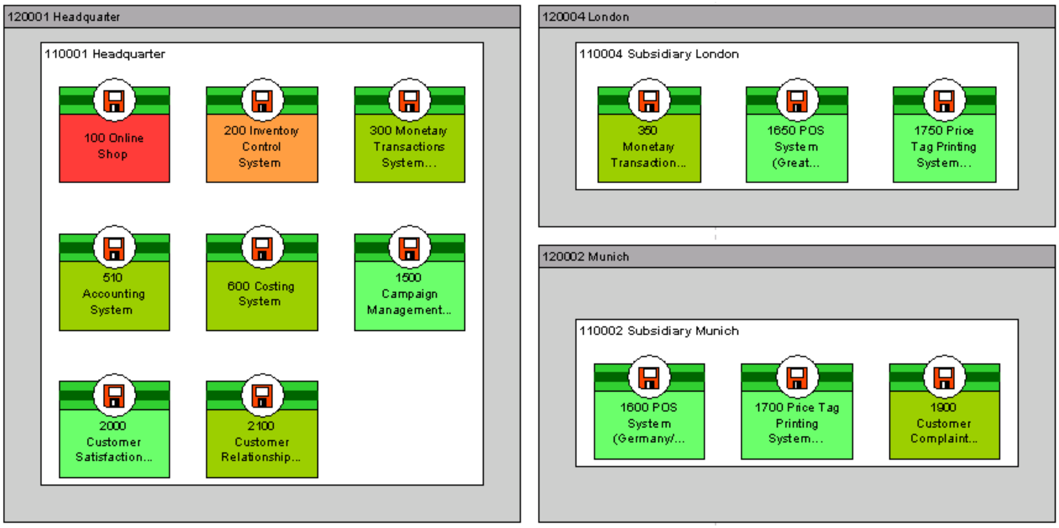


Figure 7.9.: *ADOit* - axis Interacting with, Editing of, and Annotating Visualizations: Application systems colorized based on the number of users

First, roles and user groups can be created based on the customers needs. Then, the roles can be added to the user groups and the user groups can be added to certain users, like shown in Figure 7.10. The creation of roles is very flexible, e. g. they can be set to restrict the view of the corresponding user to all visualizations within a certain category, e. g. all *infrastructure models*, or just to restrict certain attributes of a class. For example, in order to edit certain attributes of the class *goal* a role *editGoals* can be assigned to the *goals and strategy* user group. For other users, that are not in the group, the *notebook* areas are greyed out and not editable.

Furthermore, *ADOit* provides capabilities to alter dialogs based on the user's rights (grey out of attributes) or menu entries (restriction symbol).

If more users work collaboratively on the data, only one user at a time is allowed to edit data. Thus, data elements that are to be changed will be locked. Hereby, locking is possible on model, object, and attribute level.

Further functionalities of *ADOit* are the creation of histories, basic workflow management, and notification capabilities that will be detailed as follows.

For each object meta data will be automatically stored, regarding information, like creation date, last modification date, or last modifying user. Furthermore, meta data will be automatically stored for visualizations, like author, last modification date, modifying users, or number of objects and relations in the visualization.

The tool provides a basic workflow management. For each visualization a status can be set to *draft*, *quality assurance*, *released*, or *archived*. In combination with a basic notification function, users can be informed, if work from their side is required, e. g. the review by another modeler. The identification of aged data or aged visualizations is not directly supported, but can be performed easily via the integrated *search* function.

If implemented, versioning of elements is supported. Therefore, a model can be reverted to a status prior to changes by a certain user.

ADOit supports various languages and also provides the capability to assign a locale to an user profile. Also adaptations to e. g. date format, or currency are possible. Furthermore, the tool is flexible regarding menu entries and context (right mouse-click) that can be customized by a skilled user by

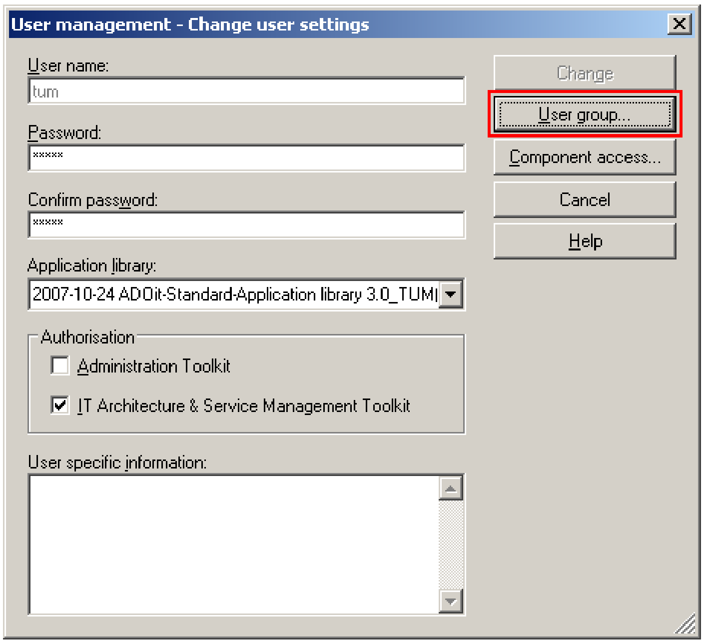
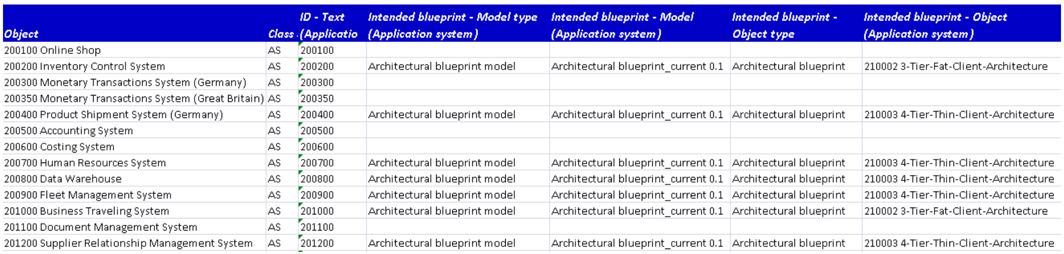


Figure 7.10.: *ADOit* - axis Communication and Collaboration Support: User management: Editing roles

utilizing the built-in scripting engine. However, switching between languages within one installation is not possible[1].

Regarding the requirement of editing data using an external editor, *ADOit* provides an add-on tool called *Homer*. *Homer* uses the tabular and function-oriented structure of Microsoft Excel, to support the user during the gathering or editing of data, like shown in Figure 7.11. The changes that were made utilizing *Homer*, can then be imported to (or exported from) *ADOit*.



Object	Class	ID - Text (Applicatio	Intended blueprint - Model type (Application system)	Intended blueprint - Model (Application system)	Intended blueprint - Object type	Intended blueprint - Object (Application system)
200100 Online Shop	AS	200100				
200200 Inventory Control System	AS	200200	Architectural blueprint model	Architectural blueprint_current 0.1	Architectural blueprint	210002 3-Tier-Fat-Client-Architecture
200300 Monetary Transactions System (Germany)	AS	200300				
200350 Monetary Transactions System (Great Britain)	AS	200350				
200400 Product Shipment System (Germany)	AS	200400	Architectural blueprint model	Architectural blueprint_current 0.1	Architectural blueprint	210003 4-Tier-Thin-Client-Architecture
200500 Accounting System	AS	200500				
200600 Costing System	AS	200600				
200700 Human Resources System	AS	200700	Architectural blueprint model	Architectural blueprint_current 0.1	Architectural blueprint	210003 4-Tier-Thin-Client-Architecture
200800 Data Warehouse	AS	200800	Architectural blueprint model	Architectural blueprint_current 0.1	Architectural blueprint	210003 4-Tier-Thin-Client-Architecture
200900 Fleet Management System	AS	200900	Architectural blueprint model	Architectural blueprint_current 0.1	Architectural blueprint	210003 4-Tier-Thin-Client-Architecture
201000 Business Traveling System	AS	201000	Architectural blueprint model	Architectural blueprint_current 0.1	Architectural blueprint	210002 3-Tier-Fat-Client-Architecture
201100 Document Management System	AS	201100				
201200 Supplier Relationship Management System	AS	201200	Architectural blueprint model	Architectural blueprint_current 0.1	Architectural blueprint	210003 4-Tier-Thin-Client-Architecture

Figure 7.11.: *ADOit* - axis Communication and Collaboration Support: Homer

Furthermore, *table views* containing data of all objects of a certain class (c.f. Section 7.1.1) can be transferred by copy and past to Microsoft Excel, where changes can be made and then transferred back by copy and paste to *ADOit*. This is a very convenient way, using the full editing features provided by Microsoft Excel, however the user should take care and attention in using this feature to avoid errors arising from copy and paste.

ADOit provides a *HTML report*, for exporting visualizations and reports. During the creation of HTML files, the user can decide which models, classes, and attributes should be part of the report.

[1]According to *BOC* is planned to be implemented in the next version.

Furthermore, this HTML report provides additional features, like selecting the attributes, which should be provided, zooming of the visualization, or which further relationships to other objects should be provided.

In order to ease the communication and gathering of information concerning the EA management endeavour, *BOC* provides the *ADOit web client*, which is the lightweight access component. Here, it is possible to get live access to reports and visualizations through a web client, like shown in Figure 7.12. Like in the client, the rights management is here available as well.

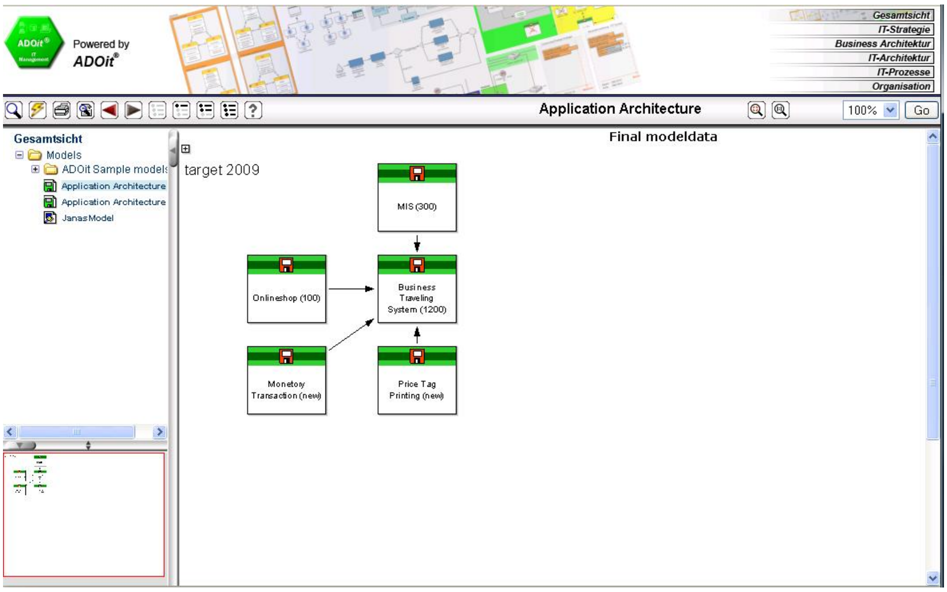


Figure 7.12.: *ADOit* - axis Communication and Collaboration Support: *ADOit web client*

For visualizations features like zoom, or clickable regions in visualizations are available. Between visualizations, and the *table view*, a structured table, containing all objects of the visualization, can be switched. The *notebook* for each object can be accessed either through the visualization or through the *table view*. In the notebook the attributes of objects can be edited. However, further editing regarding visualizations, or introducing new attributes is not available.

ADOit web client provides a search function and query component that searches for objects as well as for names of visualizations or reports. Furthermore, a navigation is available, that allows filtering options for visualizations based on the folder structure. Besides, a *helicopter view* for visualizations allows the user to go around the visualization quite easily. The style of *ADOit web client*, e. g. another background color, can be adapted. Further customization can be made by *BOC* consultants.

Rating: 4

7.1.5. Flexibility of the Information Model

ADOit standard configuration is shipped with an out of the box information model. The information model contains over 50 classes for EA management and is derived from the EA management frameworks *TOGAF* and *Zachman Framework*. In addition, the information model is highly adaptable. Therefore, it is possible to create, change, hide, or delete classes and their attributes as well as relationships. This can be done utilizing the *ADOit administration toolkit*. Furthermore, it is possible to have more than one information model within one installation of *ADOit*. For example, for this survey, the *ADOit standard configuration* as well as the *ADOit for SoCaStore* were available within one installation.

Classes can be created based on customers needs. For defining the symbolic representation of classes a flexible scripting language called *GraphRep* is available. It allows the implementation of a symbol for each class that can also be annotated, based on the value of attributes (like described in Section 7.1.3. Additionally, it is possible to delete classes for which objects are contained in the database, after a warning is displayed. Therefore, to avoid a loss of data, it is recommended to export the information model (called *application library*), which is to be modified, as a safety rule.

Furthermore, *ADOit* provides some functionality regarding attributes. As already mentioned, it is possible to add new attributes for new as well as for existing classes. Therefore, a variety of predefined data types, e. g. string, integer, listbox, date, or formula are available. Furthermore, it is possible to use regular expressions to define *ranges* for certain attribute values. Besides, it is possible to define default values for attributes. Figure 7.13 shows the creation of a new attribute for the newly created class *SLA*. Deleting attributes that already have assigned values is not allowed. As mentioned above, these relationships should be hidden, to avoid inconsistencies in the modeled scenarios.

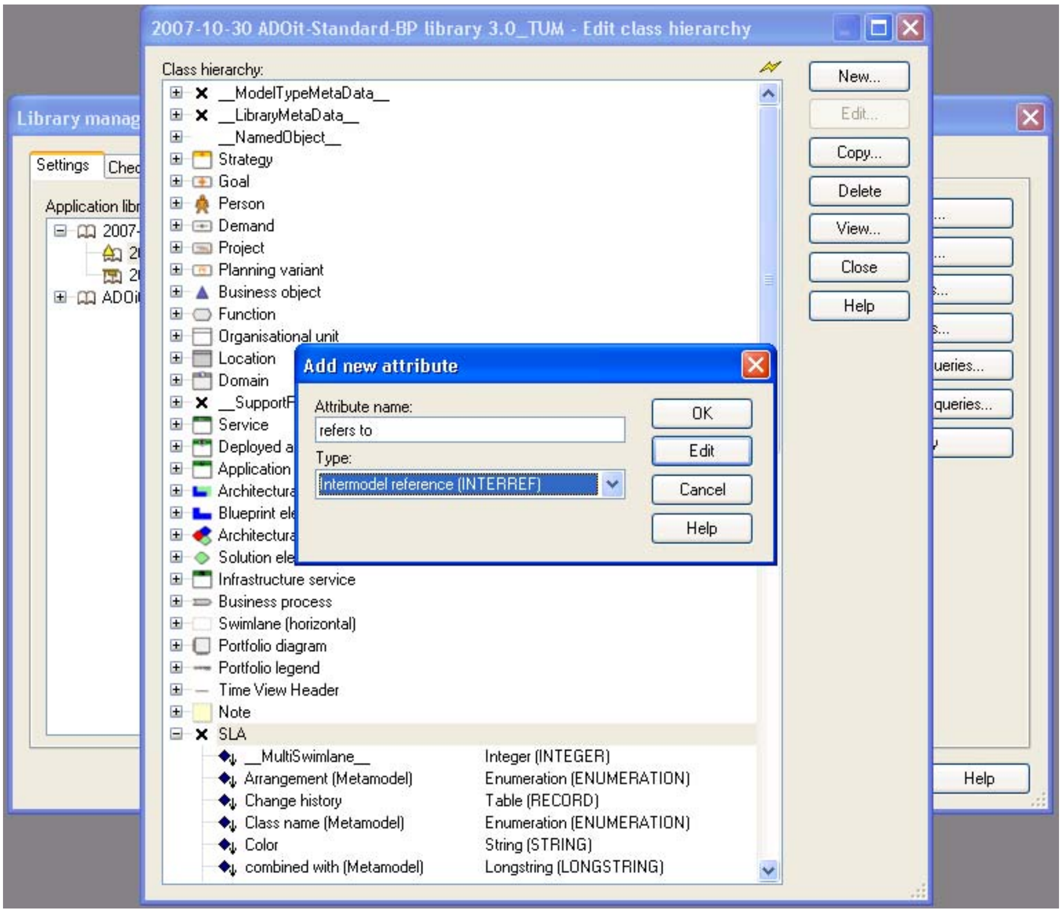


Figure 7.13.: *ADOit* - axis Flexibility of the Information Model: Adding a new attribute to the information model

Predefined relations can be adapted, hidden, and deleted, as well as new relationships can be created. Deleting relations that have already been used is not allowed. To avoid inconsistencies within the modeled scenarios these relations should only be hidden.

The predefined information model could be adapted to cover all concepts of the *SoCaStore* information model. Furthermore, the whole information model can be exported to a text file containing all classes with their attributes and relationships. However, there exists no graphical representation of the information model.

Rating: 6

7.1.6. Support of large scale Data

ADOit provides a standard import function based on a scripting language as described in Section 7.1.1. The import of 1000, 5000, or even 10000 application systems can be excecuted in a reasonable time. For importing additional features like, importing of objects that are not already available, are provided. However, using the option *pre- or suffix* adding new syllables to objects slows the import considerably down, as each reference of the object has to be updated as well.

The opening of the *table view* as well as of the *graphical* view is considerably fast. In the *helicopter view* all application systems can be previewed quite fast. However, the visualization is much longer than broad, e. g. with 1000 application systems only five are shown in each row. Nevertheless, it can be configured, how many objects are in a row, or some application systems can be selected and moved manually, like done for Figure 7.14.

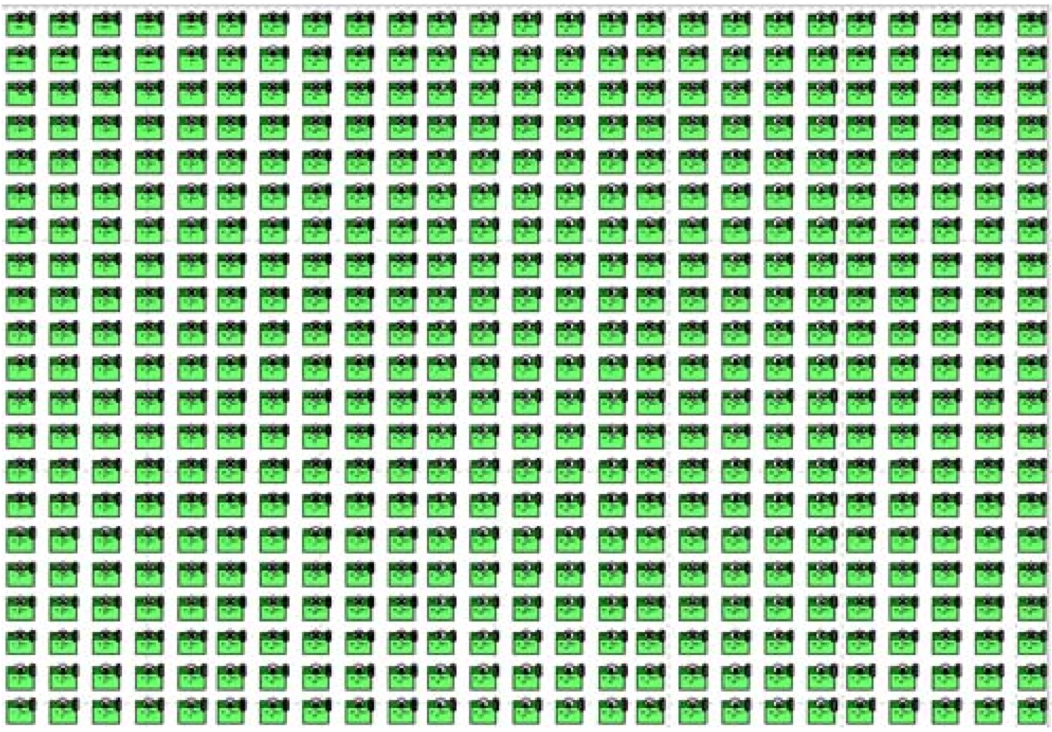


Figure 7.14.: *ADOit* - axis Support of large scale Data: Cutout of a visualization containing 5000 application systems

As far as automated visualizations are supported, the data from the repository can be visualized, although a slower performance has to be accepted. Once, the visualization is created, the user can adapt it and also the adding of predefined annotations is not visibly affected.

For working with large amounts of data, the dataset can be divided through the order structure, e. g. all application systems can be separated regarding their functionality, like financial, or logistical application systems. Thus, it is easier, to find an application system or to select an application system for a visualization.

In order to facilitate the handling of large visualizations, *ADOit* provides the feature of sub models to hide a complex inner structure. Sub models provide *domains* that can be opened or closed, loading the data only on demand. Therefore, it allows to open referenced models quickly. Furthermore, visualizations can be structured hierarchically. Thus, by clicking on a hierarchically section, another visualization can be opened, that e. g. provides more details. This functionality can be used, e. g. for expanding the ICT infrastructure of a certain subsidiary or to expand the software architecture of an application system.

Rating: 6

7.1.7. Impact Analysis and Reporting

ADOit standard configuration provides different predefined relation tables for creating reports, e. g. on infrastructure elements and their installed software components. In addition, impact analyses, called *graphical analyses* that automatically create visualizations on the data contained in the repository, are provided. Most of the graphical analyses are flexible e. g. regarding the selected class or regarding the objects within the class that should be analyzed. For example, a swimlane diagram can be created from one selected application system, showing its IT services and processes. Furthermore, graphical analysis can be newly created or existing ones can be adapted. Therefore, a scripting language is available through *ADOit administration toolkit*.

For *ADOit for SoCaStore* several graphical analyses were created, like the one shown in Figure 7.15. For example Figure 7.28 shows an automatically created swimlane diagram containing strategy, goal, demand, and application systems. During the creation of the visualization the user decides which strategies should be considered in the report. Latter, all relationships to the other objects will be provided. Figure 7.16 shows a portfolio matrix visualizing application systems regarding their criticality, number of modifying users, and number of projects. These examples could be further adapted utilizing the scripting language, e. g. changing the colors, or the shown attributes. Further graphical analyses can be created utilizing the customizing features of *ADOit*.

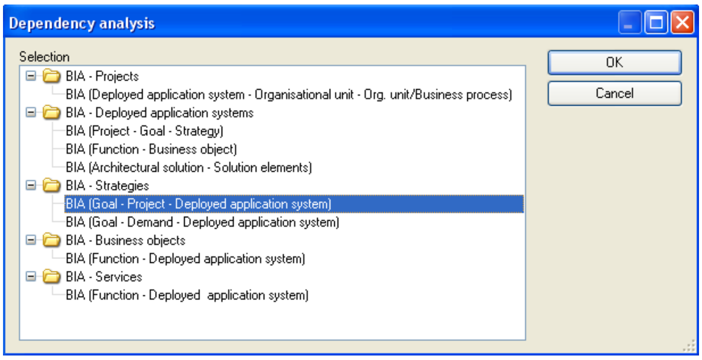


Figure 7.15.: *ADOit* - axis Impact Analysis and Reporting: Selecting graphical analyses

Rating: 5

7.1.8. Usability

ADOit consists of two main applications the client *ADOit IT Architecture and Service Management Toolkit* and the *ADOit administration toolkit*, and several add-on tools, like *Homer*.

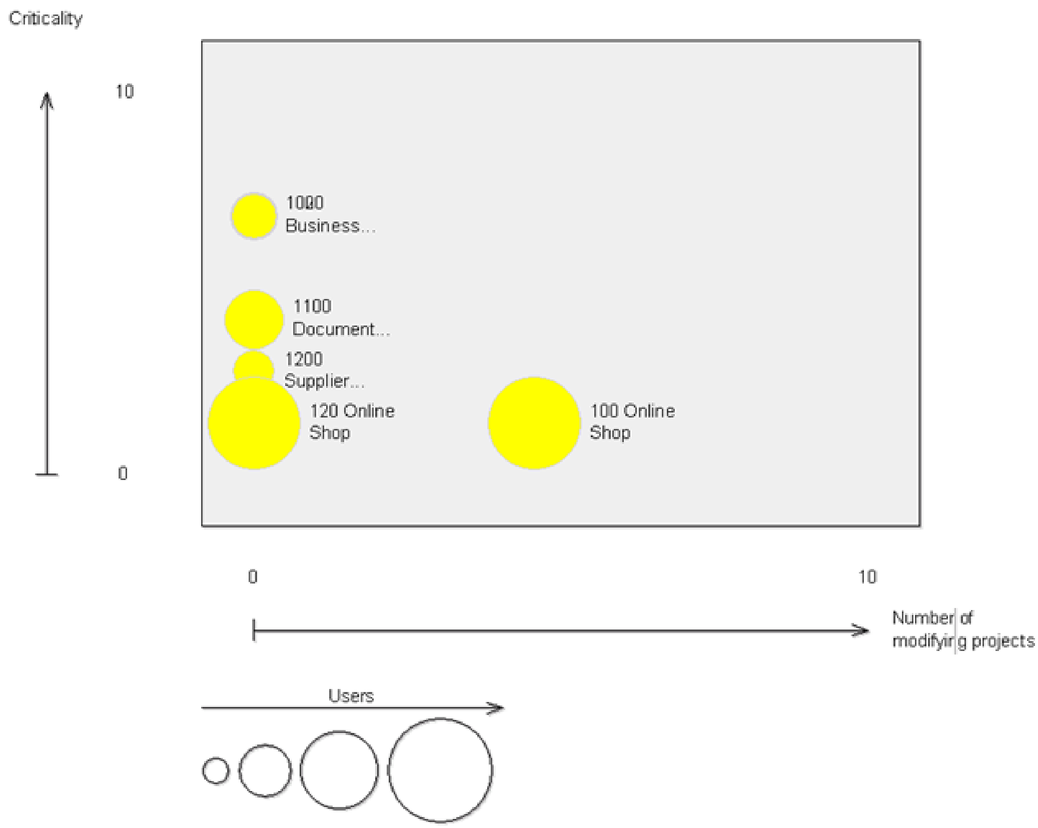


Figure 7.16.: *ADOit* - axis Impact Analysis and Reporting: Portfolio matrix visualizing application systems regarding their criticality, number of modifying users, and number of projects

The handling of the client is intuitive. It offers a well-structured user interface providing access to the different components, like *modeling* and *analysis* that can be easily switched. Furthermore, the client offers a well structured menu, action icons, and right mouse-click functions that are context sensitive. The client is split up in seperate windows. Especially, a little window on the left side, showing a *helicopter view* of the entire visualization and containing a clickable region to navigate to another part of the visualization, is helpful. Furthermore, the client provides functionality, like search, and filtering mechanisms. The tool also provides a comprehensive help functionality with the sections contents, index, and find.

Because of the *BOC* approach to provide a meta tool, *ADOit* is highly customizable. *ADOit standard configuration* can be shipped with or without active repository. If shipped with active repository, *ADOit* provides a very flexible information model in which classes, relationships, and attributes can be created, edited, or deleted. Furthermore, automatically generated visualizations and reports can be created based on the customer needs through a scripting language called *ADOscript* that can be utilized by *BOC* consultants, but could be used by a skilled customer. Additionally, menu and right mouse-click functionality can be adapted and enhanced.

Regarding the connection between once created visualizations and the objects, relationships, and attributes in the data in the information model two different approaches are available. *ADOit standard configuration* treats visualizations as *snapshots* that represent a certain point of time of the application landscape. If needed, changes can be done, but are not transferred to the repository. However, *ADOit* could be adapted, in a way, that visualizations are connected to the objects in the repository and therefore transferred to the repository.

The handling of the *ADOit administration toolkit* is intuitive as well. Like the client, it is structured in different components, e. g. *user management*, *library management*, and *component management* that can be easily switched. However, the same user cannot be logged on at the *ADOit IT Architecture and Service Management Toolkit* and the *ADOit administration toolkit* at the same time.

Rating: 5

7.2. Evaluation of EA Management Support

The evaluation of the EA management support refers to the adapted *ADOit for SoCaStore* version provided by *BOC* consultants. Most of the functionality is part of *ADOit standard configuration*, if not pointed out differently. Especially, most automatically created visualizations and reports are added for the *SoCaStore* using the configuration and customizing features of *ADOit*.

7.2.1. Landscape Management

This scenario consists of two parts. On the one hand, it requires visualization and report functionality regarding the current application landscape, on the other hand, it requires the ability to model future states for planned and target application landscapes.

ADOit provides functionalities for organizing current, planned, and target landscapes via a folder structure, like shown in Figure 7.17. The folder structure can be created manually by the customer and is therefore adaptable, e. g. the application landscape can be further structured regarding the different EA management tasks that should be covered. Figure 7.17 illustrates the integration of the sebis structure integrated in *ADOit*. By clicking on a certain layer, the appropriate objects can be selected, to create a visualization for that specific task.

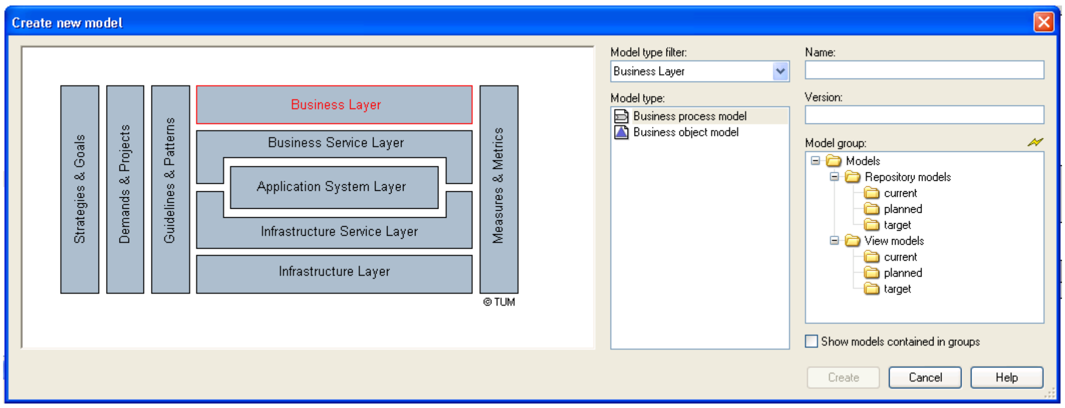


Figure 7.17.: *ADOit* - axis Landscape Management: Structure of the applications landscape

The tool allows to handle several variants of the application landscape, because it is possible to store an arbitrary number of variants of an object within the repository, e. g a new object can be created with an attribute that specifies it as a future state object. Thus, it is possible to create provided, configured or customized visualizations and reports for each state, as usually the objects that should be within a visualization, can be selected before the creation.

Like described in section 7.1.2 once a visualization in *ADOit for SoCaStore* is created, it is disconnected from the repository and cannot be updated[2]. This mechanism allows persisting and archiving visualizations as a visualization represents a certain state of the repository at creation time. If future states should be visualized, two alternatives exist: On one hand, a visualization can be created with objects from the repository that belong to a future state (see above), on the other hand, a once created visualization can be graphically adapted. Due to the fact that the visualizations are not connected to the repository, the changes made are just visible in the visualization, and not represented in the repository. However, if modifications should latter be propagated to the repository, the changes can be manually made within the repository and subsequently be referenced by the visualization object through the visualization objects notebook.

ADOit provides a comparison facility for comparing two visualizations of the same type. Therefore, different types of comparisons are provided. The first type compares objects and connectors. Thus, it is checked, if all objects of one visualization are also displayed in the second visualization. It is also checked, if all objects are connected to each other in the same way. The second type compares objects and connectors including their attributes. If there are any variations the object or the connector will be highlighted, like shown in Figure 7.18. Due to the fact that the comparison function checks the name of objects and relationships, as well as attribute values (if requested), the functionality does not cover graphical changes, e. g. like resizing of an object. A third comparison variant is provided for the *table view*, which contains data of all objects of a certain type (see Section 7.1.1). The, therefore, available comparison function compares the attributes of objects within a table view.

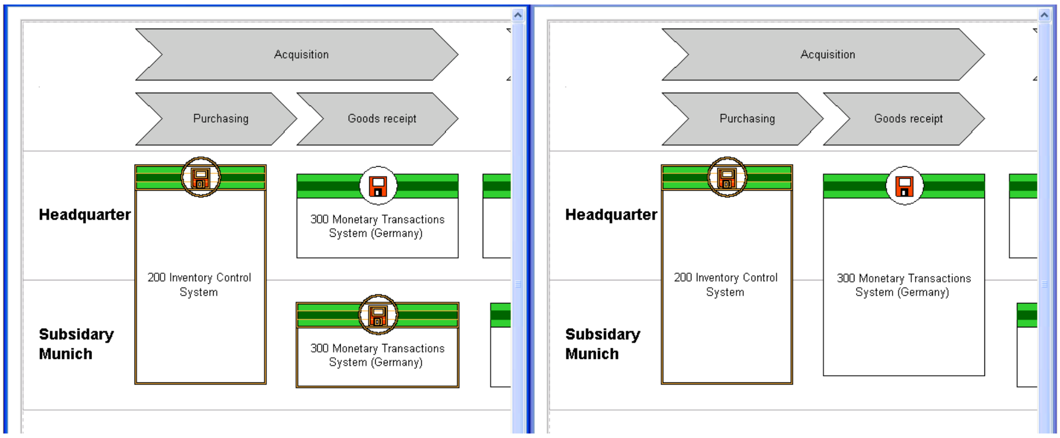


Figure 7.18.: *ADOit* - axis Landscape Management: Comparison of the application landscape

Furthermore, *ADOit* provides another way to handle variants of models. Within one visualization it is possible to handle several variants of this visualization and switch between those. For example Figure 7.19 shows one variant for the *target 2009*. However, this functionality is limited, as for e. g. the comparison function described above cannot be used here.

In *ADOit* it is possible to historize the reasons for the changes made on objects or visualizations. Therefore, an attribute called *change history* exists for every object, where changes can be manually recorded.

The procedure consistency is mostly given, as current and future states of the application landscape can be visualized, but visualizations are treated as *snapshots* and in order to visualize changes in the application landscape a new visualization has to be created, where all manually performed graphically changes are not available. Furthermore, there exists a little limit, as only two visualizations can be compared at one time.

[2]A connection of visualizations to the repository could be customized by *BOC consultants*

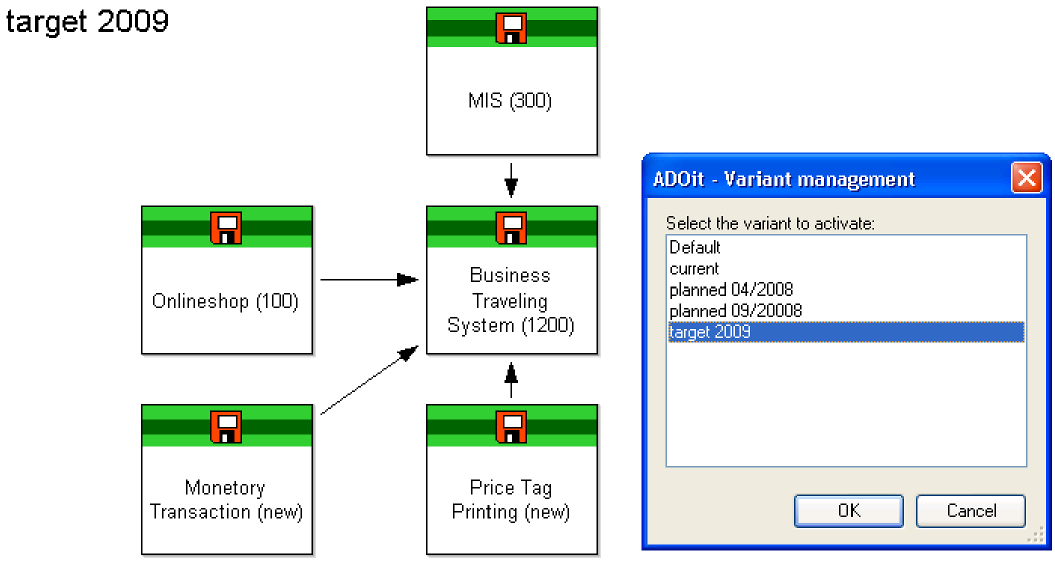


Figure 7.19.: *ADOit*- axis Landscape Management: Variant Management

The procedure integration is completely given.

Rating: 6

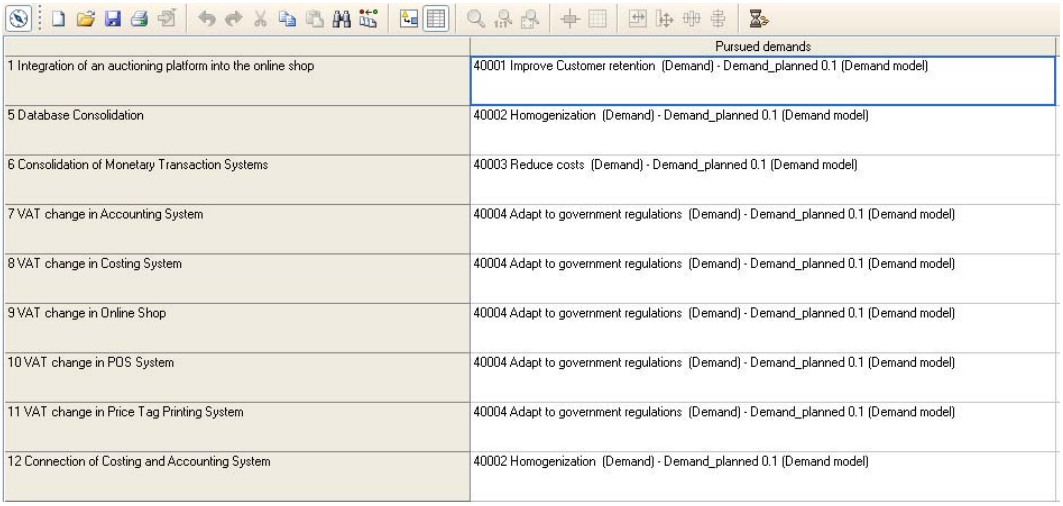
7.2.2. Demand Management

The class *demand* does not belong to the standard information model of *ADOit*, but can be created easily with appropriate attributes and relationships through the *ADOit administration toolkit*. After that, all demands can be imported and can be used for reports and / or visualizations. Therefore, questions like: which demands exist, which attributes do they have, and which relationships to other objects do they have, can be answered.

Within the *ADOit for SoCaStore* version of *ADOit*, a demand is derived from a goal. Therefore, visualizations can be created, e. g. a graphlayout map can be automatically generated for one demand showing all supported goals. Also a customized swimlane diagram can be created automatically, that visualizes the relationships between a strategy, a goal, and a demand, like shown in Figure 7.28.

Furthermore, it is possible to maintain a connection between demands and resulting projects. For this, a customized, automatically created report can be provided showing demands with their derived projects, like shown in Figure 7.20. In addition, further reports can be specified by a skilled user utilizing the integrated scripting language. However, if new projects result from a demand, the projects have to be added, creating new datasets and entering the data manually. It is not possible, to directly create a project reusing information of the respective demand.

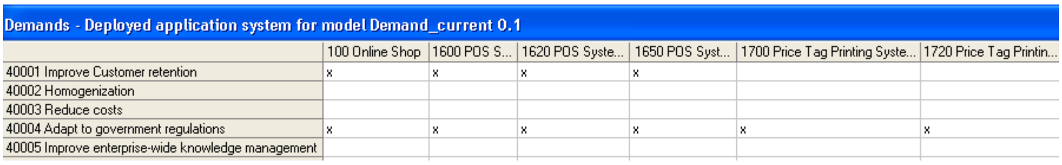
The relationship between demands and the application systems affected by them, can be visualized as well. For example a customized report can be created showing demands with their related application systems like shown in Figure 7.21. The report has been created utilizing *relation tables*, where all the relations between the chosen demand objects and affected application systems are shown. Another way analyzing the relations between demands and application systems is again to create a graphlayout map. Therefore, a demand object can be selected and the related objects (here application system) can be chosen. However, a graphlayout map can only be created for one demand at a time. If more than one demand object should be selected, a swimlane map can be customized utilizing the



	Pursued demands
1 Integration of an auctioning platform into the online shop	40001 Improve Customer retention (Demand) - Demand_planned 0.1 (Demand model)
5 Database Consolidation	40002 Homogenization (Demand) - Demand_planned 0.1 (Demand model)
6 Consolidation of Monetary Transaction Systems	40003 Reduce costs (Demand) - Demand_planned 0.1 (Demand model)
7 VAT change in Accounting System	40004 Adapt to government regulations (Demand) - Demand_planned 0.1 (Demand model)
8 VAT change in Costing System	40004 Adapt to government regulations (Demand) - Demand_planned 0.1 (Demand model)
9 VAT change in Online Shop	40004 Adapt to government regulations (Demand) - Demand_planned 0.1 (Demand model)
10 VAT change in POS System	40004 Adapt to government regulations (Demand) - Demand_planned 0.1 (Demand model)
11 VAT change in Price Tag Printing System	40004 Adapt to government regulations (Demand) - Demand_planned 0.1 (Demand model)
12 Connection of Costing and Accounting System	40002 Homogenization (Demand) - Demand_planned 0.1 (Demand model)

Figure 7.20.: *ADOit* - axis Demand Management: Demands and their supporting projects

customization features of *ADOit*. Once a visualization is created it can be annotated with further information like outlined in Section 7.1.3. Figure 7.8 for example provides a cutout for the demand *Adapt to Government* with annotated application systems.



Demands - Deployed application system for model Demand_current 0.1

	100 Online Shop	1600 POS S...	1620 POS Syste...	1650 POS Syst...	1700 Price Tag Printing Syste...	1720 Price Tag Printin...
40001 Improve Customer retention	x	x	x	x		
40002 Homogenization						
40003 Reduce costs						
40004 Adapt to government regulations	x	x	x	x	x	x
40005 Improve enterprise-wide knowledge management						

Figure 7.21.: *ADOit* - axis Demand Management: Demands with their related application systems

The procedure consistency is mostly given, because manual effort is required as the class *demand* is not provided by the predefined information model, but can be introduced by customization.

The procedure integration is mostly given. Information previously entered into the tool can be reused, although no direct transformation from demands to projects is supported.

Rating: 5

7.2.3. Project Portfolio Management

The *ADOit standard configuration* information model provides a class called *project*, which is defined as a temporary organization, that is created for the purpose of delivering one or more business products according to a specified business case. For this class several attributes like *valid from* (validity for the planned beginning of the project), *valid until* (planned date for the end of the project), *priority* (very high, high, middle, low, and very low), and *status* (open, work in progress, postponed, stopped, and closed) are available that are helpful for this scenario. Further attributes, like *urgency*, *criticality*, and *pursued demands* can be created utilizing the *ADOit administration toolkit*. Furthermore, for every project object an attribute to reference the current, planned, and target landscape can be filled. Via this reference the connection between projects and application landscapes can be shown.

In addition, all relationships required by *SoCaStore* can be created. Once relationships are available, several swimlane diagram can be defined utilizing the integrated scripting language, which traverses through the information model to collect the dependencies as requested. Once the respective swimlane diagrams are created, they can perform the required impact analyses of the scenario, as e. g. traversing the relationships between projects, application systems, business processes, and organizational units. Figure 7.22 shows a swimlane diagram with projects and their affected application systems.

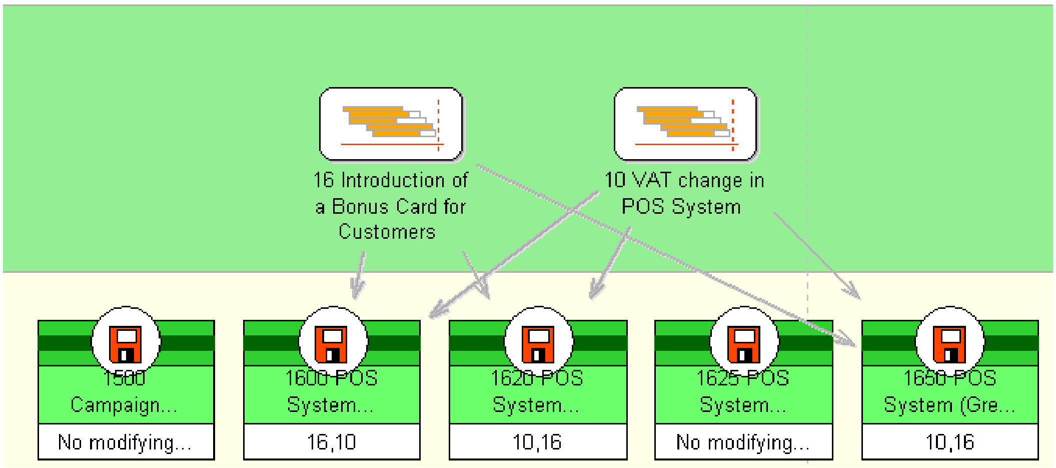


Figure 7.22.: *ADOit* - axis Project Portfolio Management: Projects with the affected application systems

By customization, this visualization is additionally annotated with the information of modifying projects providing their ID. This annotation is available for all models, visualizing application systems. However, if switching visualizations the annotation has to be added again, as the project id is only determined for the present moment. Further, if a user makes changes in the project model e. g. by referencing new application systems, these changes would not be considered in the visualization. Therefore, the visualization has to be created new or further customization of the *ADOit for SoCaStore* is required.

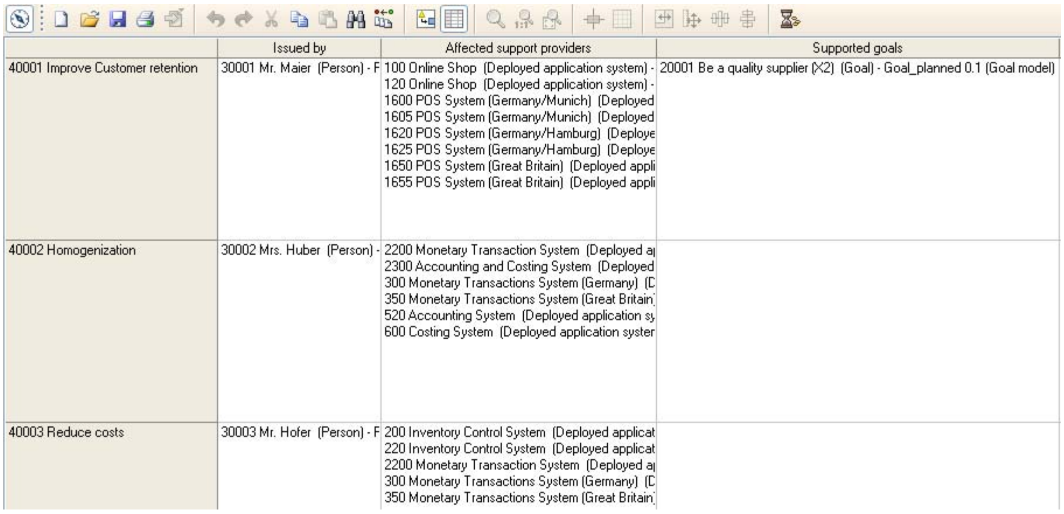
Portfolio matrices can be created via a configuration file. Figure 7.6 shows a visualization displaying application systems as bubbles, application system size as the bubble radius, application system criticality rating on the y-axis, and a number of project proposals affecting the application systems on the x-axis.

To realize the demanded reports traversing is required, which can be done utilizing the integrated scripting language. Once the reports have been defined, they can be performed on currently available objects of the classes they were created for. Figure 7.23 provides projects with information regarding the responsible person, affected application systems and their supported goals. Hereby, it is possible to have multiple information within one cell, like shown for application systems.

Furthermore, a customized relationship report can be created, like shown in Figure 7.24. This report shows not only, if there could be potential project conflicts, but provides the time period, the affected business processes and the affected application systems.

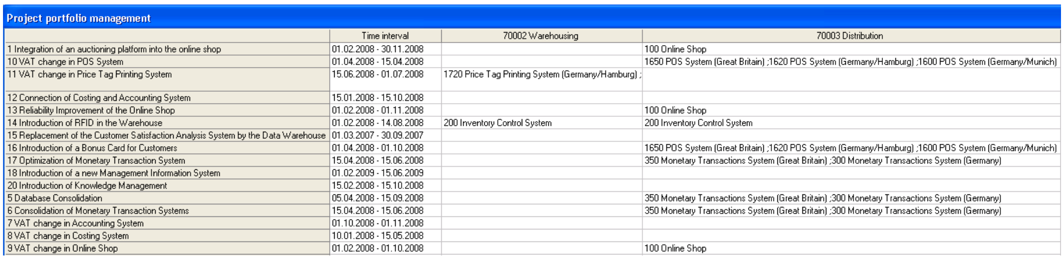
ADOit also provides interfaces to support the exchange of data with project management tools, like Mercury PPM, BW (SAP), MS Project or ADOscore.

The procedure consistency is mostly given, as further customization, e. g of the information model, is required to support the scenario.



	Issued by	Affected support providers	Supported goals
40001 Improve Customer retention	30001 Mr. Maier (Person) - F	100 Online Shop (Deployed application system) - 120 Online Shop (Deployed application system) - 1600 POS System (Germany/Munich) (Deployed 1605 POS System (Germany/Munich) (Deployed 1620 POS System (Germany/Hamburg) (Deploye 1625 POS System (Germany/Hamburg) (Deploye 1650 POS System (Great Britain) (Deployed appli 1655 POS System (Great Britain) (Deployed appli	20001 Be a quality supplier (X2) (Goal) - Goal_planned 0.1 (Goal model)
40002 Homogenization	30002 Mrs. Huber (Person) -	2200 Monetary Transaction System (Deployed a 2300 Accounting and Costing System (Deployed 300 Monetary Transactions System (Germany) (C 350 Monetary Transactions System (Great Britain 520 Accounting System (Deployed application sy 600 Costing System (Deployed application syster	
40003 Reduce costs	30003 Mr. Hofer (Person) - F	200 Inventory Control System (Deployed applicat 220 Inventory Control System (Deployed applicat 2200 Monetary Transaction System (Deployed a 300 Monetary Transactions System (Germany) (C 350 Monetary Transactions System (Great Britain	

Figure 7.23.: *ADOit* - axis Project Portfolio Management: Projects with their supporting goals and their affected application systems



Project portfolio management

	Time interval	70002 Warehousing	70003 Distribution
1 Integration of an auctioning platform into the online shop	01.02.2008 - 30.11.2008		100 Online Shop
10 VAT change in POS System	01.04.2008 - 15.04.2008		1650 POS System (Great Britain) ;1620 POS System (Germany/Hamburg) ;1600 POS System (Germany/Munich)
11 VAT change in Price Tag Printing System	15.06.2008 - 01.07.2008	1720 Price Tag Printing System (Germany/Hamburg) ;	
12 Connection of Costing and Accounting System	15.01.2008 - 15.10.2008		
13 Reliability Improvement of the Online Shop	01.02.2008 - 01.11.2008		100 Online Shop
14 Introduction of RFID in the Warehouse	01.02.2008 - 14.08.2008	200 Inventory Control System	200 Inventory Control System
15 Replacement of the Customer Satisfaction Analysis System by the Data Warehouse	01.03.2007 - 30.09.2007		
16 Introduction of a Bonus Card for Customers	01.04.2008 - 01.10.2008		1650 POS System (Great Britain) ;1620 POS System (Germany/Hamburg) ;1600 POS System (Germany/Munich)
17 Optimization of Monetary Transaction System	15.04.2008 - 15.06.2008		350 Monetary Transactions System (Great Britain) ;300 Monetary Transactions System (Germany)
18 Introduction of a new Management Information System	01.02.2009 - 15.06.2009		
20 Introduction of Knowledge Management	15.02.2008 - 15.10.2008		
5 Database Consolidation	05.04.2008 - 15.09.2008		350 Monetary Transactions System (Great Britain) ;300 Monetary Transactions System (Germany)
6 Consolidation of Monetary Transaction Systems	15.04.2008 - 15.06.2008		350 Monetary Transactions System (Great Britain) ;300 Monetary Transactions System (Germany)
7 VAT change in Accounting System	01.10.2008 - 01.11.2008		
8 VAT change in Costing System	10.01.2008 - 15.05.2008		
9 VAT change in Online Shop	01.02.2008 - 01.10.2008		100 Online Shop

Figure 7.24.: *ADOit* - axis Project Portfolio Management: Projects and processes and their affected application systems

The procedure integration is mostly given, as information previously entered into the tool can be reused, although in the application landscape management created planned landscapes cannot be directly used as input for new projects.

Rating: 4

7.2.4. Synchronization Management

ADOit provides several features to support synchronization management. The class *project* can be enhanced regarding time frame and delay information. After that, time interval maps can be created manually or automatically utilizing the integrated language.

Therefore, utilizing the configuration and customization of *ADOit* most requirements of the scenario can be fulfilled. For example, it is possible to create a time interval map which shows the organizational units, which are affected by projects like shown in Figure 7.25. This automated visualization provided in *ADOit for SoCaStore* is flexible regarding the choice of the displayed organizational units. This customization was configured by using the basic *GANTT view generation mechanism* provided by *ADOit* and the traversing functionality utilizing a script language.

Furthermore, *ADOit for SoCaStore* supports the creation of a time interval map to evaluate which dependencies exist among projects. For example it can be customized, that all projects with their lifecycle can be listed. For this feature, the *project lifecycle stages* can be utilized. The delay of a project is not displayed in the visualization, but can be added via customization of *ADOit*. An active warning about interferences between projects is not provided, but within the *analysis component* queries for checking interferences are realised for *ADOit for SoCaStore*.

Another useful visualization that is supported by *ADOit for SoCaStore* shows projects along with their referenced application systems like shown in Figure 7.26. This visualization also uses color-coding regarding performed tasks by a project to the corresponding application system, like *application system is planned*, *application system will be developed*, *project will be active*, or *application system will be retired*.

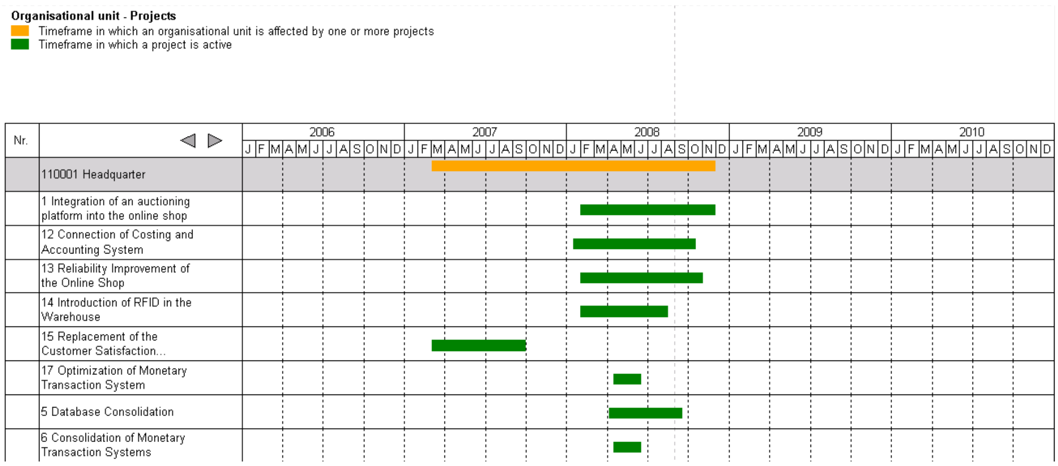


Figure 7.25.: *ADOit* - axis Synchronization Management: Time interval map with organizational units and related projects

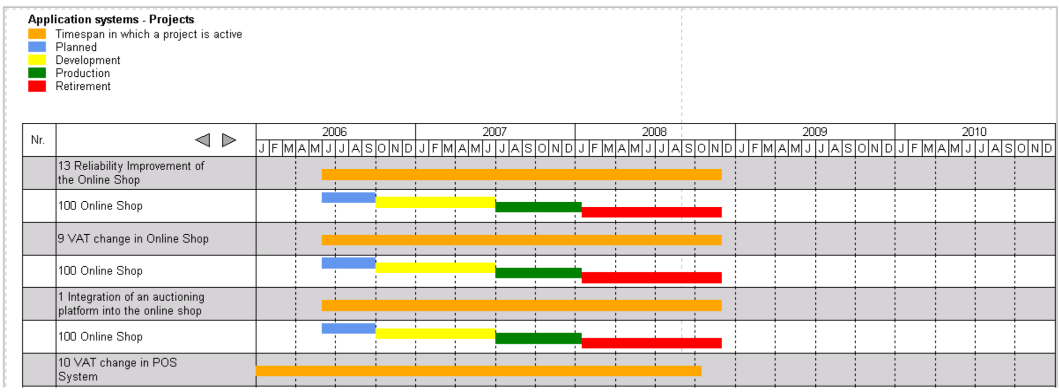


Figure 7.26.: *ADOit* - axis Synchronization Management: Time interval map with application systems and related projects

The procedure consistency is mostly given, as most requirements can be accomplished utilizing the customizing features of *ADOit*. If a delay of a project occurs, warnings are not provided online. However, reports for checking against inconsistencies are available via configured predefined analysis.

The procedure integration is mostly given, as information previously entered into the tool can be reused but an out-of-the-box support of delays is missing.

Rating: 3

7.2.5. Strategies and Goals Management

The standard information model of *ADOit* contains concepts for strategies and goals. A strategy is premeditated and defined as a long term plan of action designed to achieve a particular goal, whereas the goals of the organization describe in greater detail what should be accomplished. To support the objectives of the survey, for both classes further attributes have to be added, which was done utilizing *ADOit administration toolkit*.

ADOit supports the creation of tabular reports with *table views* that display attributes, based on the users selection, for all objects of a given class. Figure 7.27 provides a part of a *table view* for goals, which also shows the current state of a goal's fulfilment.

	Name	Description	Urgency	Reached
⊖ 1. Goal_current 0.1				
20000 Be cost effective (X1)	20000 Be cost effective (X1)	Sum of personel, material and capital cost (1)	high-middle	1.599700
20001 Be a quality supplier (X2)	20001 Be a quality supplier (X2)	Average customer satisfaction according to a performed survey (2)	middle	1.000000
20002 Reduce storage time (A1)	20002 Reduce storage time (A1)	Average time in storage (for an article) in days (3)	low-middle	2.000000
20003 Reduce time to delivery (B1)	20003 Reduce time to delivery (B1)	Maximum delivery time in days (4)	high-middle	0.555600
20004 Increase number of purchases (C1)	20004 Increase number of purchases (C1)	Number of purchases per day (6)	high	0.714300
20005 Reduce Complaints (D1)	20005 Reduce Complaints (D1)	Number of complaints per day (7)	high-middle	0.500000
20006 Increase customer satisfaction (D2)	20006 Increase customer satisfaction (D2)	Average customer satisfaction according to a performed survey (8)	high	1.000000

Figure 7.27.: *ADOit* - axis Strategies and Goals Management: Goal Properties

Furthermore, reports can be automatically created, utilizing the query builder or the configurable reporting capabilities of *ADOit*. Thus, the dependencies between goals and strategies or between projects and goals can be reflected like shown in Figure 7.23. Additionally, questions like, which organizational units have not reached their goals and by which percent, can be answered by performing a corresponding report.

Automatically generated swimlane diagrams can be created as well, utilizing the configuration and customization features of *ADOit*. Figure 7.28 provides an example, which visualizes the relationship between strategy, goal, demand, and application system. This visualization is flexible regarding the selection of the strategy objects on the first layer, and subsequently only objects that have relationships with the selected objects will be visualized. Also, a swimlane diagram can be created that traces the relations from a project up to the related strategies.

The procedure consistency is mostly given as all objectives can be accomplished, utilizing the customizing features of *ADOit*.

The procedure integration is completely given as data previously entered or visualizations and reports previously created, e. g for demand management, can be reused.

Rating: 5

7.2.6. Business Object Management

The business objects management can be performed using the standard features of *ADOit standard configuration* and the provided customizing capabilities. The standard information model provides classes and relationships that support business object management. One example is *process*, which is used for structuring the business processes of the customer, which need to be supported through

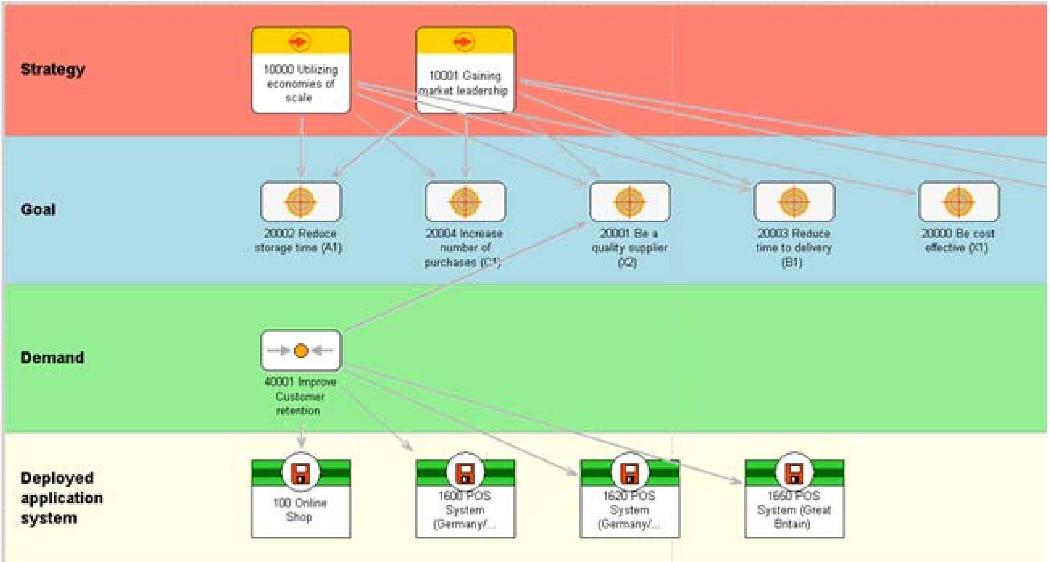


Figure 7.28.: *ADOit* - axis Strategies and Goals Management: Relationship between strategy, goal, demand, and application system

IT services and application systems. In *ADOit for SoCaStore*, the class *process* was enhanced by the appropriate attributes and new classes where created.

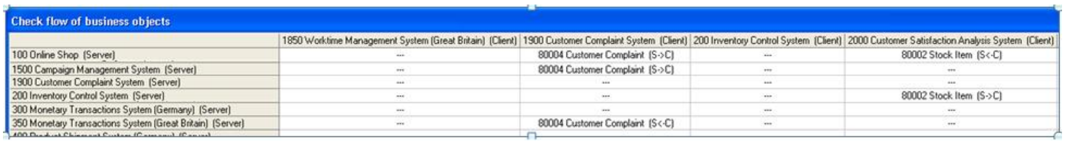
In *ADOit for SoCaStore*, a report like shown in Figure 7.29 can be created, to determine which application system uses or holds the master copy of which business object. Therefore, two attributes of the type boolean (*uses*, *master*) were added to the *business objects* class. However, no direct support for checks to identify inconsistencies are supported, e. g. if two or more application systems hold the master copy or if they hold the master copy, although they do not use the object. Nevertheless, customized reports can be created answering these questions.

	80001 Price Tag	80003 Monetary Transaction	80006 Invoice	80002 Stock Item	80005 Customer	80004 Customer Complaint
100 Online Shop	Used/Master	--	--	--	--	--
200 Inventory Control System	--	--	Used	--	--	--
2000 Customer Satisfaction Analysis System	--	--	--	--	--	--
2020 Customer Satisfaction Analysis System	--	--	--	--	--	--
2100 Customer Relationship Management System	--	--	--	--	--	--
220 Inventory Control System	Used/Master	Used	--	Used	--	--
2200 Monetary Transaction System	--	--	--	--	--	--
2300 Accounting and Costing System	--	--	--	--	--	--
2400 Knowledge Managment System	--	--	--	--	--	--
300 Monetary Transactions System (Germany)	--	--	--	--	--	--
350 Monetary Transactions System (Great Britain)	--	--	--	--	Used	Master
400 Product Shipment System (Germany)	--	--	--	--	Used	--

Figure 7.29.: *ADOit* - axis Business Object Management: Usage of business objects

The requirement, which business processes need additional manual operations in their execution, can be provided, by adding a new attribute *manual process* to the business object class and providing a report containing this attribute. Furthermore, the requirements of this scenario include the kind of operations performed in a specific application system. A report providing business objects and the operations performed on them (read, create, update, or delete) by application systems can also be created, utilizing the customizing capabilities of the tool.

The customized *ADOit for SoCaStore* supports the creation of a report that shows the flow of business objects between application systems like shown in Figure 7.30. Furthermore, it supports the manual modeling of a visualization that displays the information flow representing the exchanged business objects for specific business processes. However, the manual modeling includes some effort, as first, symbols for *business object* and *application system* are added to the canvas, and then in a second step, these symbols can be filled with the information of a specific object from the repository. Due to the fact that the visualization itself is not connected with the repository, the information flow between objects can be created in a way that does not represent the information contained in the repository.



Check flow of business objects

	1850 Worktime Management System (Great Britain) (Client)	1900 Customer Complaint System (Client)	200 Inventory Control System (Client)	2000 Customer Satisfaction Analysis System (Client)
100 Online Shop (Server)	---	80004 Customer Complaint (S->C)	---	80002 Stock Item (S<-C)
1500 Campaign Management System (Server)	---	80004 Customer Complaint (S->C)	---	---
1900 Customer Complaint System (Server)	---	---	---	---
200 Inventory Control System (Server)	---	---	---	80002 Stock Item (S->C)
300 Monetary Transactions System (Germany) (Server)	---	---	---	---
350 Monetary Transactions System (Great Britain) (Server)	---	80004 Customer Complaint (S<-C)	---	---

Figure 7.30.: *ADOit* - axis Business Object Management: Flow of business objects between application systems

The procedure consistency is mostly given, utilizing the customizing capabilities of *ADOit*.

The procedure integration is completely given, as data previously entered can be reused in the simulation of this scenario.

Rating: 5

7.2.7. SOA Transformation

For this scenario, on the one hand, profound knowledge of the current application landscape is required, to derive possible SOA transformations, on the other hand, the ability to model the performed changes has to be supported. *ADOit* provides several functionalities to support the application landscape management like described in (Section 7.2.1).

The customized *ADOit for SoCaStore* provides a relationship between *function* and *service*. Therefore, functionalities that are applicable candidates for the transformation into a business service can be added to the appropriate business service. Figure 7.31 shows a cutout of the *notebook* of a service object, displaying the supported functions.

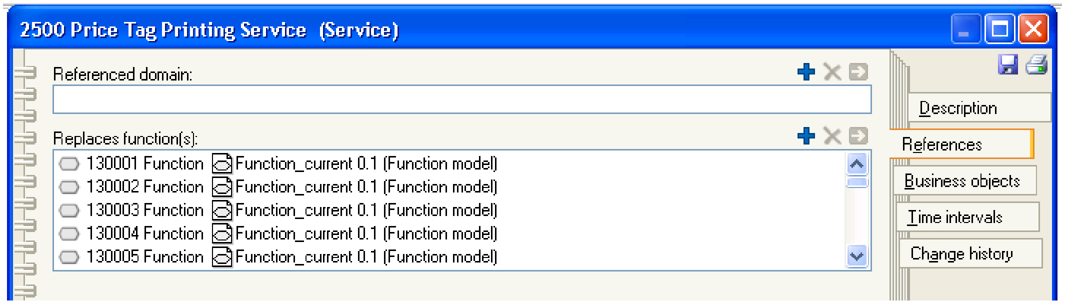


Figure 7.31.: *ADOit* - axis SOA Transformation: Relationship between functions and services

Furthermore, applicable candidates of application systems should be derived, based on information, like the number of domains, or the number of processes using the application. This objective can be reached using the customizable annotation functionality provided by *ADOit*. Figure 7.32 visualizes a manually created process support map that can automatically be colorized regarding the attribute *number of users*. Thus, potential SOA transformation candidates can be derived. To evaluate what service level certain business services fulfill, an attribute, e. g. called *SLA*, can be defined. After the objects are assigned the respective values, as e. g. 24h / 7d, the information can be derived, either via the *table view* displaying the *SLA* attribute, or utilizing the *standard query* for subsets of data.

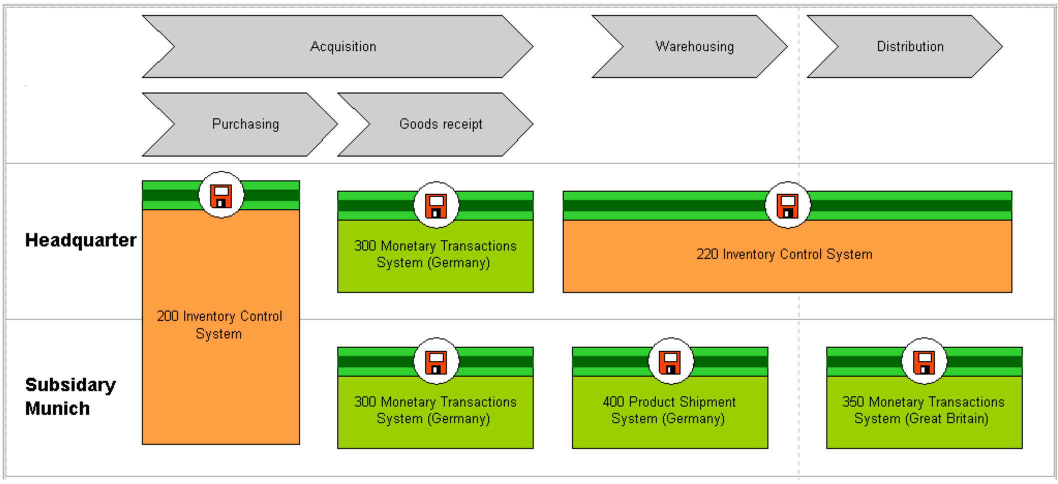


Figure 7.32.: *ADOit* - axis SOA Transformation: Selecting appropriate application systems

To analyze the interrelation of business services, functions, and application systems, a swimlane diagram can be created, utilizing the configuration and customizing features of *ADOit*, like done for the relationship between strategy, goal, demand, and application system shown in Figure 7.28.

Furthermore, several standard reports are provided. For example a report, which lists the business services and their service level agreements, can be derived opening the *tabular view* for the *business service* objects. Thus, all business services are listed with their name, ID, description, and service level agreement. Additional reports can be created, utilizing the customizing features of *ADOit*. However, a time interval map showing the migration of functionalities from an application system to a business services is not directly supported.

The procedure consistency is mostly given, as most of the objectives can be achieved utilizing the customization features of *ADOit*.

The procedure integration is mostly given, as once created visualizations cannot be further used.

Rating: 5

7.2.8. IT Architecture Management

The IT Architecture Management deals with the introduction and implementation of architectural solutions standardizing the architecture of specific application systems to address heterogeneous application landscapes. To address this scenario, new classes and their relationships, *blueprint element*, *solution element*, *architectural blueprint*, and *architectural solutions* with their attributes and relationships can be added to the information model. After that, visualizations and reports can be created to support this scenario, using the customizing features of *ADOit*.

For example a visualization can be created highlighting standard application systems, like Figure 7.33, that shows an automatically created cluster map customized for SoCaStore, where standard application systems are annotated with a thicker boarder. This visualization can further be annotated showing the architectural solution of the application system. To meet the requirement, which application system use which technologies a swimlane diagram can be created, that visualizes the technology for all chosen application systems.

Reports can be further created, to support this scenario. For example a report can be created, that shows which architectural solution should be maintained and which should be replaced. Therefore,

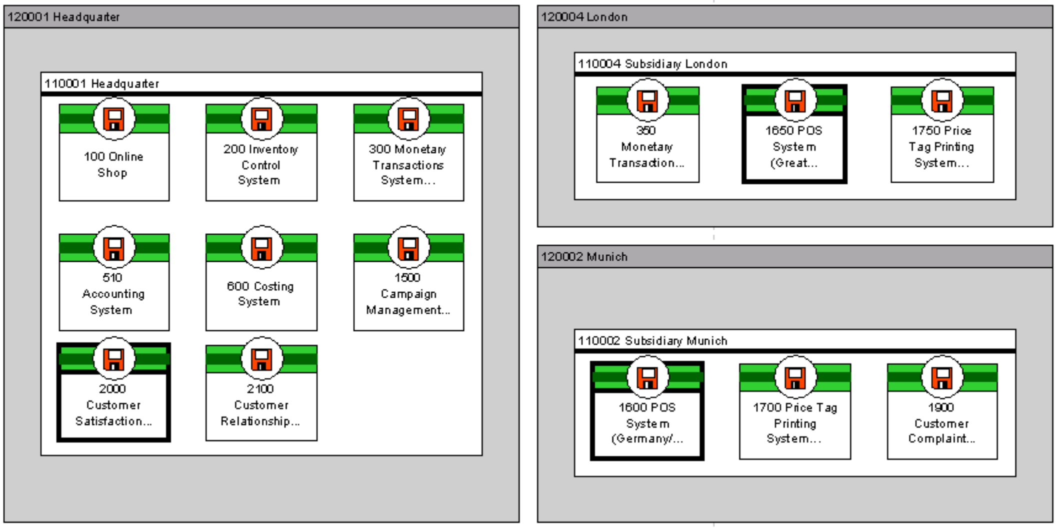


Figure 7.33.: *ADOit* - axis IT Architecture Management: Cluster map with annotated standard application systems

the user can enter the *target state* of an architectural solution, which is a flag attribute indicating, if the architectural solution will be maintained or replaced. Also, a report can be created, that shows only replaced or maintained architectures. This can be done utilizing the standard query features of *ADOit*. Furthermore, another report, like shown in Figure 7.34, can be created, that provides the architectural solutions and the number of conforming application systems. This report is calculated by first entering each application system object and then traversing through the repository collecting all the solution elements that make up an architectural solution.

Architectural solution	No. of conforming Deployed application systems
230001 OracleTomcat	12
230002 OracleBea	2
230003 OracleFatClient	8
230004 BeaFatClient	6
230005 MySQLTomcat	5
230006 DB2BeaFatClient	2
230007 MySQLBeaFatClient	7

Figure 7.34.: *ADOit* - axis IT Architecture Management: Architectural solutions and the number of conforming application systems

The procedure consistency is mostly given, as the objectives can be achieved utilizing the customizing features of *ADOit*.

The procedure integration is completely given as data previously entered can be reused.

Rating: 6

7.2.9. Infrastructure Management

The standard information model of *ADOit* provides several classes like *infrastructure*, *database*, or virtual infrastructure element. To support the information model of *SoCaStore* the *infrastructure* class is used, where further attributes were applied.

Therefore, reports and visualizations can be created to support the *SoCaStore* scenario. For instance a standard query can be used, retrieving the information, which databases are currently in use. Similarly, the information, which databases are in danger of running out of support, can be gathered through the *table view* like shown in Figure 7.35.

	Name	ID	Description	Type	Start production	End production	Support end date
⊖ 1. Solution element_current 0.1							
240001 MySQL 2.1	240001 MySQL 2.1	240001		not defined	2005:10:01	2009:06:15	2010:12:31
240002 Tomcat 5.1	240002 Tomcat 5.1	240002		not defined	2005:10:01	2009:08:30	2010:06:01
240003 Apache 2.0.53	240003 Apache 2.0.53	240003		not defined	2005:10:01	2009:08:30	2010:12:31
240004 IE 6.0	240004 IE 6.0	240004		not defined	2006:01:01	2009:08:30	2010:07:01
240005 Oracle 9i	240005 Oracle 9i	240005		not defined	2006:01:01	2013:12:31	2015:01:31
240006 Bea Weblogic 8.1	240006 Bea Weblogic 8.1	240006		not defined	2006:01:01	2009:12:15	2010:12:31
240007 Proprietary Fat-Client	240007 Proprietary Fat-Client	240007		not defined	2006:01:01	2010:03:31	2011:05:01

Figure 7.35.: *ADOit* - axis Infrastructure Management: Solution elements and their attributes

To visualize, which application system uses which database a swimlane diagram can be created, that visualizes the relationship between infrastructure and application system objects. Furthermore, to show in the visualization only the databases and not all other infrastructure elements, e. g. web server, the standard hiding function can be utilized, like down for Figure 7.36. This approach can also be used, showing only databases that are planned to be replaced and the therefore effected application systems.

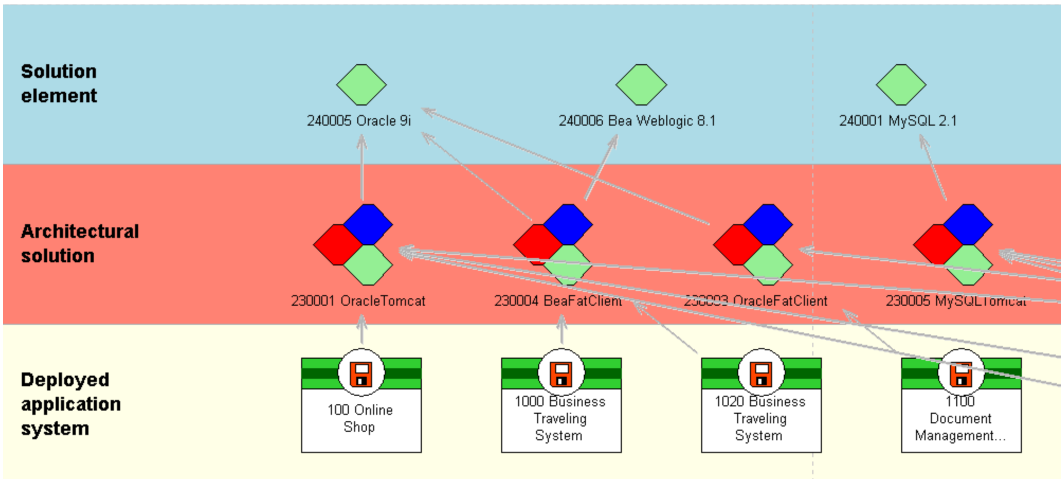


Figure 7.36.: *ADOit* - axis Infrastructure Management: Swimlane diagram for databases and their supported application systems

The procedure consistency is mostly given, as the objectives can be achieved utilizing the customizing features of *ADOit*.

The procedure integration is completely given as data previously entered can be reused.

Rating: 4

7.3. Tool Vendor's Profile

ADOit- Architecture and Service Management

Mag. Christoph Moser (Product Manager)

One of the main challenges for today's IT managers is the strict alignment of information technology with the company's business objectives. Thus, creating more opportunities to co-operate with corporate management and to support their business processes using IT gains more and more importance. As a result, extended requirements on quality, availability and costs of IT must be met. Implementing an effective IT Architecture and Service Management solution accomplishes these.

ADOit® *supports IT Management through the integration of ITIL, COBIT and CMMI into a comprehensive IT Governance framework. Enterprise Architecture Management (EAM) is supported through modelling and analysis of 'as-is', 'planning' and 'target' scenarios.*

Metamodelling Concepts of *ADOit*

Due to the integral view from the strategic level right down to the ICT infrastructure, the IT services are effectively and efficiently integrated into a well managed Enterprise Architecture. *ADOit* employs a flexible metamodel, which enables you to integrate the concepts of Service Management and Enterprise Architecture Management. Its standard metamodel is based upon widely accepted Enterprise Architecture and Service Management standards such as TOGAF, Zachman and ITIL.

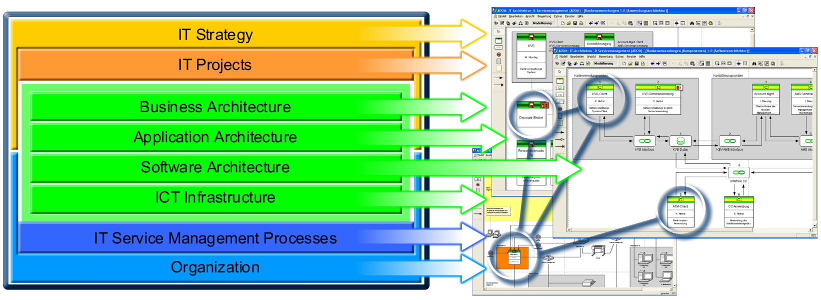


Figure 7.37.: BOCAF® Framework

One of the main advantages of *ADOit*® is its means for metamodelling. The metamodelling capabilities of *ADOit*® allow to define customer-specific information models (metamodels). Therefore, modelling classes and diagram types can be easily customised to satisfy the bespoke requirements without the need for extensive or expensive programming time.

The metamodel is surrounded by core components in *ADOit* ®. For example there is release workflow; an integrated query language and reporting mechanism. This can be used to generate views that can be easily altered by the *ADOit*® administrator. Defining a user specific method, tailored precisely according to the customer's requirements, is a primary service of BOC.

Company Profile of the BOC Group

BOC Information Technologies Consulting GmbH was established in 1995 in Vienna as a spin-off from the Business Process Management Systems (BPMS) Group from the Department of Knowledge and Business Engineering at the University of Vienna. Through its international orientation and customer

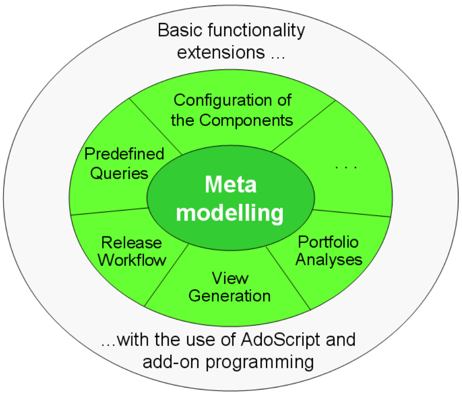


Figure 7.38.: Metamodelling Concepts of *ADOit*®

centric strategy, new branch offices in Berlin, Madrid, Dublin, Athens and Warsaw were established for local support. Today BOC can count manifold organizations on all five continents to its customers. These are individually and language specifically treated by experienced consultants and partners.

The BOC Group is a consulting and software company specialising in IT-based management solutions. The software products are used worldwide. The consulting expertise of BOC includes the following areas:

- Strategy Management with ADOscore®
- Business Process Management with ADONIS®
- Supply Chain Management with ADOlog®
- IT Architecture and Service Management with ADOit®

Additional Application Scenarios of ADOit

Besides EAM process management in accordance to worldwide accepted 'standards' like ITIL is supported. ADOit provides facilities for designing and evaluating IT services and for publishing the service catalogue. IT Governance is supported by the provision of COBIT reference models including appraisal mechanisms for the estimation of maturity levels. Through its open interfaces and flexible metamodel ADOit can be integrated with CMDBs or operate as CMDB on its own.

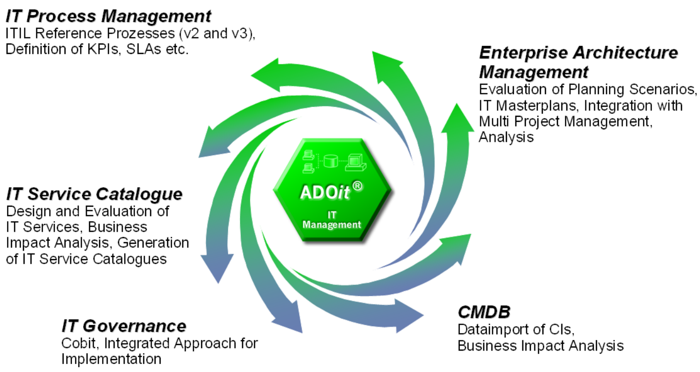


Figure 7.39.: *ADOit*® - IT Management

CHAPTER 8

Casewise (Corporate Modeler Suite & IT Architecture Accelerator)

Statement to Analysis of Corporate Modeler Suite and IT Architecture Accelerator

At the beginning of the survey, a *short list* of tools to be evaluated was compiled based on the input of the sponsors and partners (see Section 1.5). The *Casewise Corporate Modeler Suite* and the *IT Architecture Accelarator* were ranked high in this list and therefore, we tried to get in contact with the team of Casewise Ltd. in order to work out a plan for performing an evaluation of the tool.

The initial contact attempts were not successful, after a couple of requests for information, we finally received an answer. In this answer a spokesperson of Casewise Ltd. indicated, that the company could not spend resources on participating in our tool survey to that point in time. Further, an evaluation, which had been performed not long ago by a consulting company, was refered to, highlighting the positive result, the tools from *Casewise* had received there.

In response to this information, we tried to communicate the approach of our survey and emphasized, that participating in our tool survey is not likely to cause heavy investments by the tool vendor. We further detailed the way, our evaluation would be performed in. Nevertheless, this clarification did not result in actions by Casewise Ltd. and we did not receive a notification of acceptance nor rejection.

In accordance with the sponsors of the survey, we subsequently decided not to include an evaluation of the tools from Casewise in the survey, as such an evaluation would have had to be based on a trial version of the respective application. This would have led to incomparable performance results as well as a potentially limited overall set of functionalities. In contrast, we provide this short statement, giving an indication, why the *Casewise Corporate Modeler Suite* and the *IT Architecture Accelarator* were not evaluated in this survey.

CHAPTER 9

Embarcadero (EA/Studio)

Contents

9.1. Evaluation of Specific Functionality

Below, we provide the results of the evaluation of *Embarcadero EA/Studio* in regard to the scenarios for analyzing specific functionalities.

9.1.1. Importing, Editing, and Validating

Embarcadero EA/Studio can import data from different data sources, amongst others *Embarcadero ER/Studio* and Microsoft Visio. Subsequently, the capability to import data from *comma separated value (csv)* files is further evaluated in detail. A csv-file to be imported by the tool is organized to contain a instance data in every row, with the attributes from the predefined information model to be spread over the columns. Additionally, it is important, that the file only contains one object type to be imported, as this type is specified in the import dialog (cf. Figure 9.1) and thus determines, how the data contained in the file is interpreted. As part of the importing task, *Embarcadero EA/Studio* performs a validation on the data from the file in order to check, whether it is consistent with the information model or not. If consistency errors, e. g. values not compatible with the expected data type, are found, an error report like the one in Figure 9.2 is displayed.

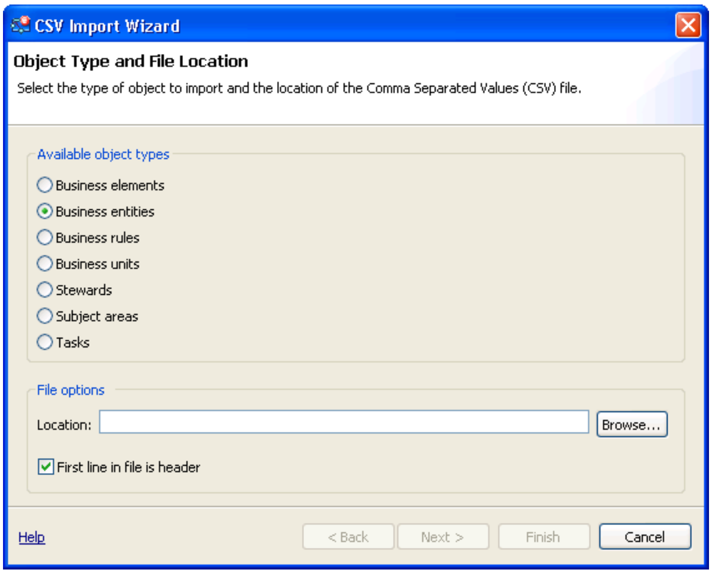


Figure 9.1.: *Embarcadero EA/Studio*- axis Importing, Editing, and Validating: Import dialog

During the import, the tool cannot perform transformation of the data to be imported, but these functionality could easily be leveraged relying on the capabilities of Microsoft Excel via which the csv-files can be handled natively. Furthermore, the import capabilities of *Embarcadero EA/Studio* are limited, as the user can import attribute values, but cannot import relationship information. This information can subsequently be entered into *Embarcadero EA/Studio* using the editing facilities of the tool. During import, *Embarcadero EA/Studio* can automatically create a visualization of the objects imported, by placing them equally distributed on the canvas of an existing or new diagram, leading to a visualization as the one shown in Section 9.1.6.

For editing data as stored in the *Embarcadero EA/Studio*, the user can utilize the so called *grid editor* (see Figure 9.3). In this editor the attribute values of multiple objects can be changed in a convient way. Additionally, the user can apply different sortings on the information displayed.

Rating: 3

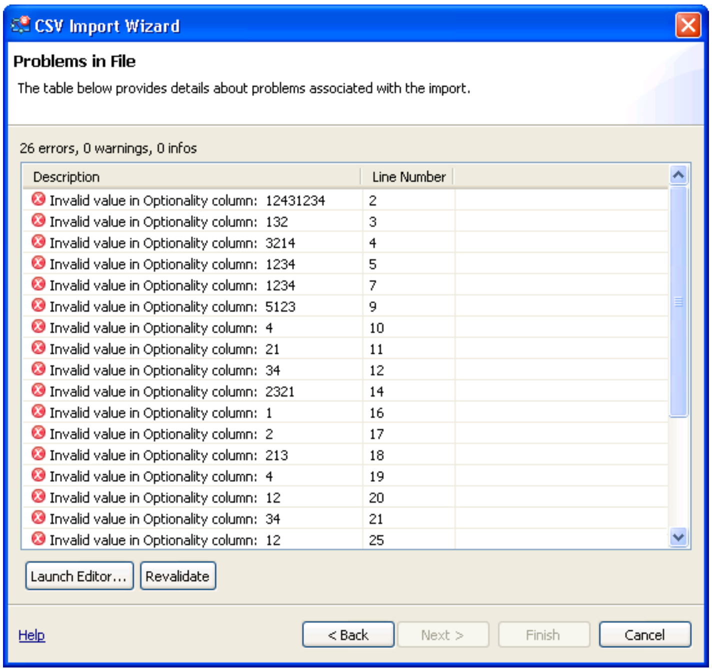


Figure 9.2.: *Embarcadero EA/Studio*- axis Importing, Editing, and Validating: Dialog highlighting errors in the import file

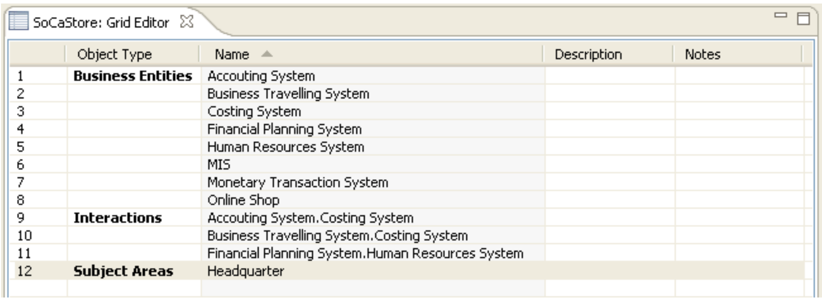


Figure 9.3.: *Embarcadero EA/Studio*- axis Importing, Editing, and Validating: Grid editor for editing multiple objects

9.1.2. Creating Visualizations

Embarcadero EA/Studio provides predefined editors for creating visualizations of the information governed by the tool. These visualizations are mainly graphlayout maps (cf. Figure 9.4) according to our terminology as described in Section 4.1.2. In addition to that, the concept of nesting symbols for expressing relationships is extensively used, such that cluster map like visualizations (see Figure 9.5) can also be created. Nevertheless, no automation for generating these visualizations is present. Concludingly, some manual interaction has to be employed in creating the visualizations, e. g. via moving and resizing the symbols displayed.

Further software maps, as e. g. the process support map or the time interval map (cf. Section 4.1.2) can also be created in *Embarcadero EA/Studio*, but employ manual interaction as no automated generation is possible. The same applies for portfolio matrices, although the functionality to export the data to a *comma separated value (csv)* file could be used here for employing a spreadsheet tool like Microsoft

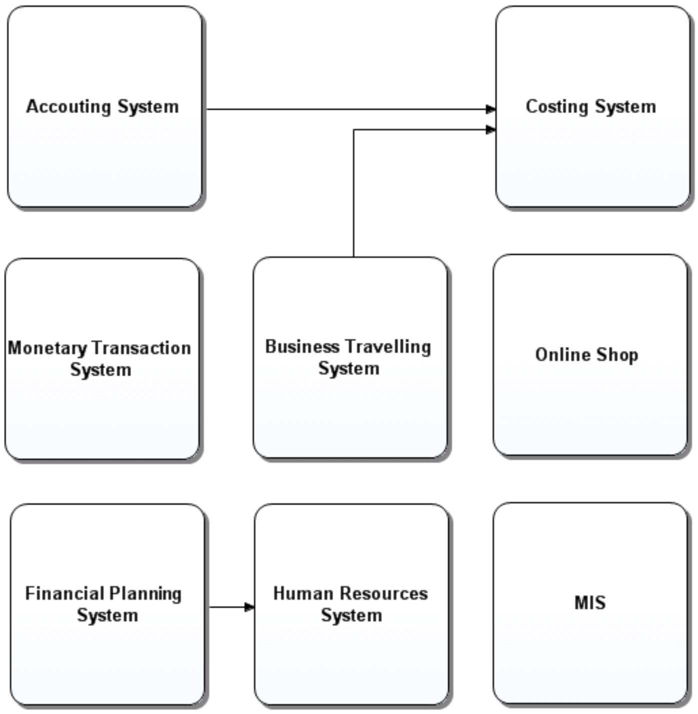


Figure 9.4.: *Embarcadero EA/Studio*- axis Creating Visualizations: Cutout of a graphlayout map

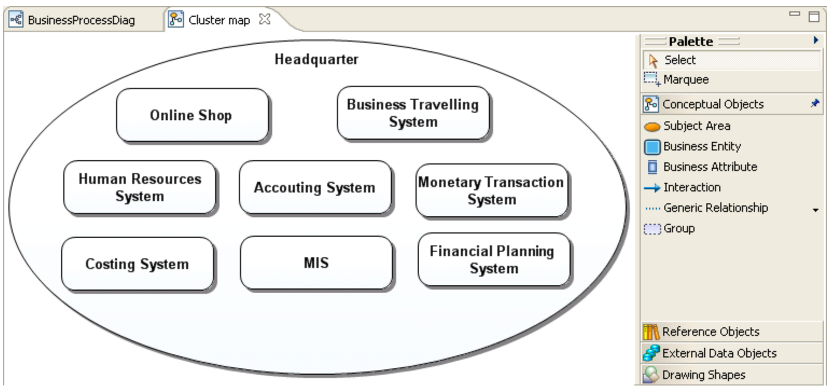


Figure 9.5.: *Embarcadero EA/Studio*- axis Creating Visualizations: Cutout of a cluster map

Excel. Finally, swimlane diagrams can be created manually using *Embarcadero EA/Studio*, leading to a visualization as shown in Figure 9.6.

To all the visualizations as alluded to above, the limitations of the information model of *Embarcadero EA/Studio* apply (see also Section 9.1.5). Therefore, only detailed business processes and business data can be modeled, while no application systems or other more abstract concepts related to EA management are available.

Rating: 2

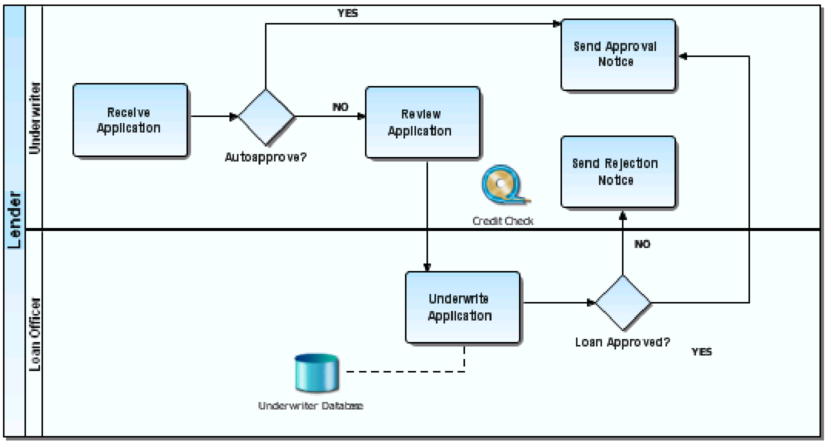


Figure 9.6.: *Embarcadero EA/Studio*- axis Creating Visualizations: Swimlane diagram

9.1.3. Interacting with, Editing of, and Annotating Visualizations

Embarcadero EA/Studio supports editing of formerly created visualizations by providing different functionalities. Firstly, the user is free to apply minor changes to the graphical makeup of a visualization, such as changing the font-size, font-face, or the text alignment. In contrast, the user cannot change the symbol representing a specific object to differ from the default symbolic representation of the corresponding class. Thereby, visual consistency is ensured. Secondly, the user can move and resize symbols freely, potentially executing semantic changes therein, if a symbol, having a semantic transported via relative positioning, is moved.

Embarcadero EA/Studio provides additional means for ensuring semantic consistency of the visualization. If the user deletes a symbol, a dialog is raised, asking, whether the corresponding element from the underlying data should also be deleted or not. Finally, the tool applies consistency checks, if used for business process modeling, e. g. the diagram is checked, if it is conformant to the BPMN or not, and the user can decide whether to allow the non-conformity or not.

In order to support the creation of visually appealing diagrams, the user can employ additional functions available from a dropdown menu to set a common alignment for a group of symbols, e. g. to the left. Further, the user can call functions, setting a group of symbols to the same height or width. Additionally, a group of symbols can automatically be distributed vertically or horizontally, providing the user thereby with basic layout support. Finally, a grid can be displayed for the diagrams and the symbols can be determined to snap to the grid.

Rating: 3

9.1.4. Communication and Collaboration Support

Embarcadero EA/Studio does not rely on a repository to store data according to its built in information model, but employs files for performing this function. These files can only be accessed via the user interface of the *Embarcadero EA/Studio*. Thus, no lightweight client for reading access is currently provided. Additionally, the tool does not provide mechanisms for supporting collaborative and concurrent work of multiple users. According to Embarcadero, such a functionality is going to be provided in a future release.

Nevertheless, *Embarcadero EA/Studio* provides support for using Microsoft Excel as an external editor for performing changes to parts of the underlying data, namely the attribute values of the objects. The objects can be exported to a csv-file, which is natively supported by Microsoft Excel and subsequently changed there. In re-importing the information, the user can choose whether to use the information from the file to update the information of the corresponding existing objects in the tool or to overwrite the data. Nevertheless, the capabilities are somewhat limited here, as conflicting changes applied both in the exported file and the user interface of *Embarcadero EA/Studio* are not discovered. The data from the tool is overwritten with the data from the external file.

Rating: 2

9.1.5. Flexibility of the Information Model

Embarcadero EA/Studio comes shipped with a small-sized information model of less than 20 different classes, that directly focuses on data modeling as well as modeling of business processes, both on a very concrete level of low abstraction. While such information can, dependent on the EA management approach taken, be seen as a part of or contributing to the information neccesary for managing the EA, some core concepts as e. g. application systems or infrastructure elements are not implemented in the information model of the tool.

In the current version of *Embarcadero EA/Studio* the information model cannot be adapted by the user, neither by introducing new classes, attributes or relationships, nor by updating or deleting existing ones. As a consequence, only the concepts governed by the predefined information model can be modeled in the tool consistently to the methodology as employed there. A minor possibility to customize the information governed in the tool is given via the *link* concept, which represents a relationship between two objects. For that concept a textual attribute *name* can be set, thus providing the possibility to discern different types of basic relationships. Nevertheless, according to the vendor, it is planned to extend the information model in a future version to more concepts employed with EA management. Additionally, the vendor plans to make functionality for configuring the information model accessible to the user.

Rating: 2

9.1.6. Support of large scale Data

Embarcadero EA/Studio provides support for large scale data, which can be imported up to 1000 objects in a reasonable amount of time, even if using the automated generation of a visualization similar to the one shown in Figure 9.7. In interacting with a visualization of that size, e. g. by moving or resizing a symbol a minor diminishment of performance is experienced. The same is true, when saving such a visualization is concerned. Other interactions with the tool are recognizably not slowed down, with the exception of the deletion of objects, which are displayed in a visualization.

When larger data sets are considered, the tool undergoes a considerable diminishment of performance, especially concerning the user interaction. This slow-down reaches its peak at 5000 objects. For such an amount of data, the import and generation of data nearly take half an our, leading to a visualization, which does not support reasonable interaction performance nor can it be saved.

Zooming and navigating through large scale visualizations is conveniently supported, especially by the built-in and navigatable *overview* thumbnail. Nevertheless, other means for reducing visualization complexity, e. g. aggregating of symbols or filtering of existing visualizations are not directly supported.

Rating: 3

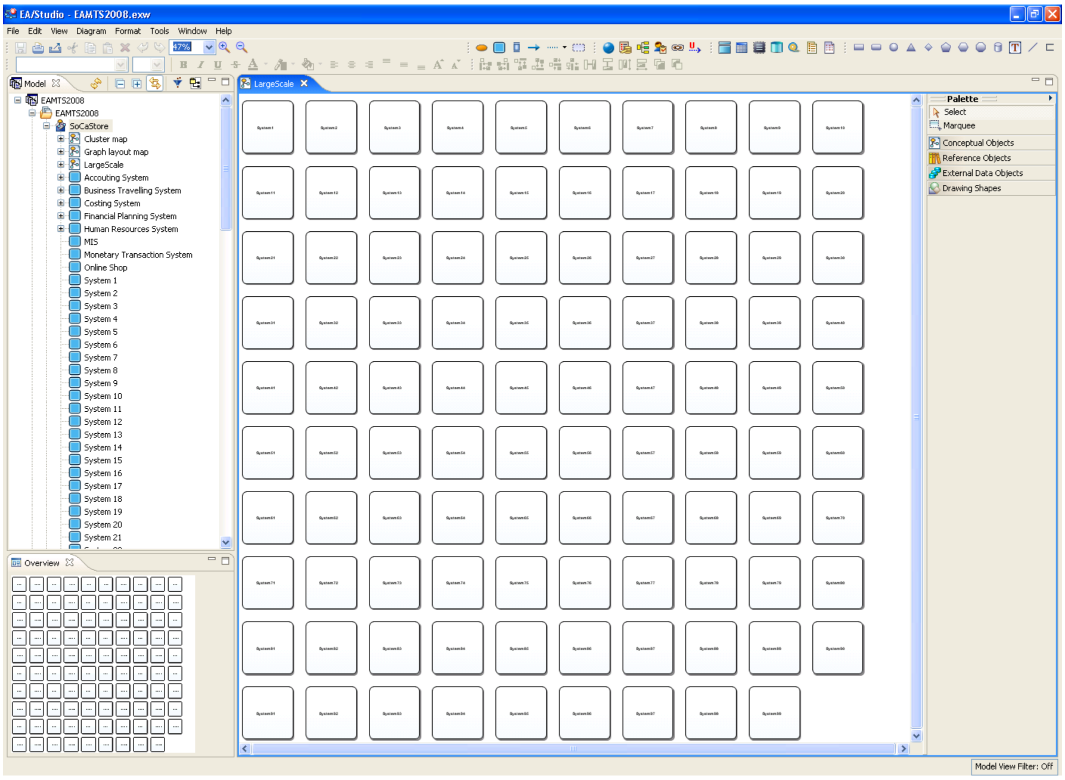


Figure 9.7.: *Embarcadero EA/Studio*- axis Support of large scale Data: Automatically created visualization of a large scale data set

9.1.7. Impact Analysis and Reporting

For performing an impact analysis, *Embarcadero EA/Studio* provides two different functionalities, namely the *Impact Analysis Diagram* and the *Impact Analysis Report*. Using one of these functionalities, the user can explore a certain object and the ones directly related. Thereby, the user is limited to one level of traversing relationships and cannot automatically perform transitive impact analyses, although such functionality could be leveraged employing some manual effort.

An *Impact Analysis Report* as shown in Figure 9.8 takes into account all the incoming and outgoing relationships of the object under consideration and displays them in a convenient, tree-like structure. On these relationships the user can apply two different kinds of filters: the first one concerning the type of relationship to be displayed and a second one selecting the classes of the related elements, which should be displayed.

In addition to the report an *Impact Analysis Diagram* can be created automatically, leading to a visualization as the one shown in Figure 9.9. Therein, the elements, whose relationships should be traversed is positioned in the center, surrounded by the related objects. In contrast to the report, not all different relationship types are considered in the generation of such a diagram - only two types of relationships, the types *link* and *usage* are taken into consideration, while other relationships from the information model cannot be utilized.

Rating: 3

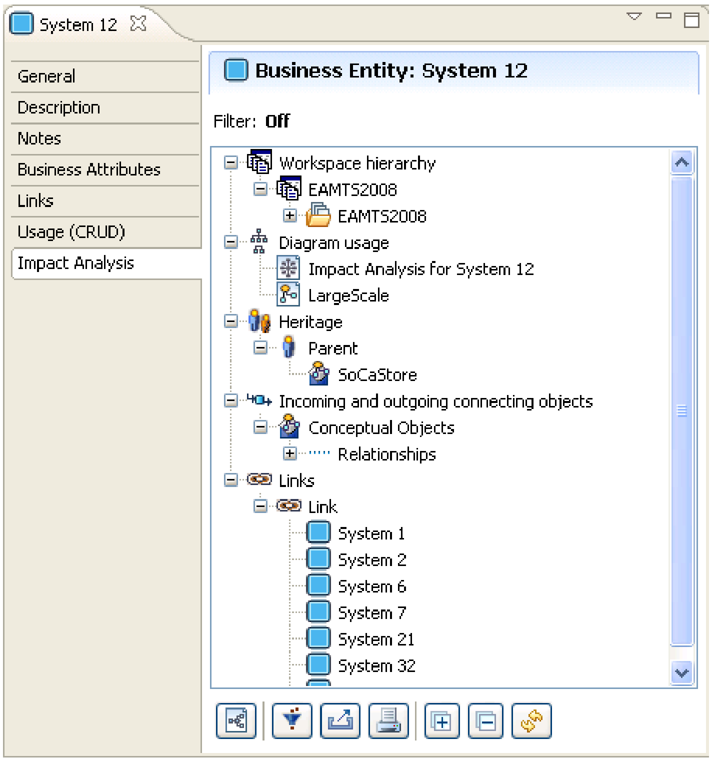


Figure 9.8.: *Embarcadero EA/Studio*- axis Impact Analysis and Reporting: Impact analysis report

9.1.8. Usability

Embarcadero EA/Studio is built as an Eclipse Rich Client application and thus leverages many of the functionalities, which are readily employed in the corresponding framework. Therefore, a user, familiar with the workspace concept as provided by many eclipse-based applications, quickly gets accustomed to the user interface as well as to the handling of the underlying data. As a consequence of the strong eclipse commitment of the tool, the user can rely on basic functionalities provided therein, especially for adapting the user interface by moving and resizing views and editors as needed.

When concerning ways for navigating and editing data governed within *Embarcadero EA/Studio*, the tool provides a number of different options, how to perform. A tree-like navigator displays the stored objects, organized according to the corresponding diagrams they are displayed on, and the classes they belong to. In this navigator, the user is provided with means to perform adaptations to the data, e. g. renaming objects or establishing new relationships. In addition to that, the user can select an object for editing its details, thereby opening another view displaying different tabs of attributes of the corresponding object. These object detail views can be stacked in any part of the user interface (cf. Figure 9.10), such that the user can make use of e. g. comparing the attribute values of two different objects.

Furthermore, the navigvation view of *Embarcadero EA/Studio* allows the user to select a filter for the objects displayed therein, so objects of a certain type or used for a certain modeling task as e. g. *data modeling* (also see Section 9.1.5) can be hidden.

Rating: 7

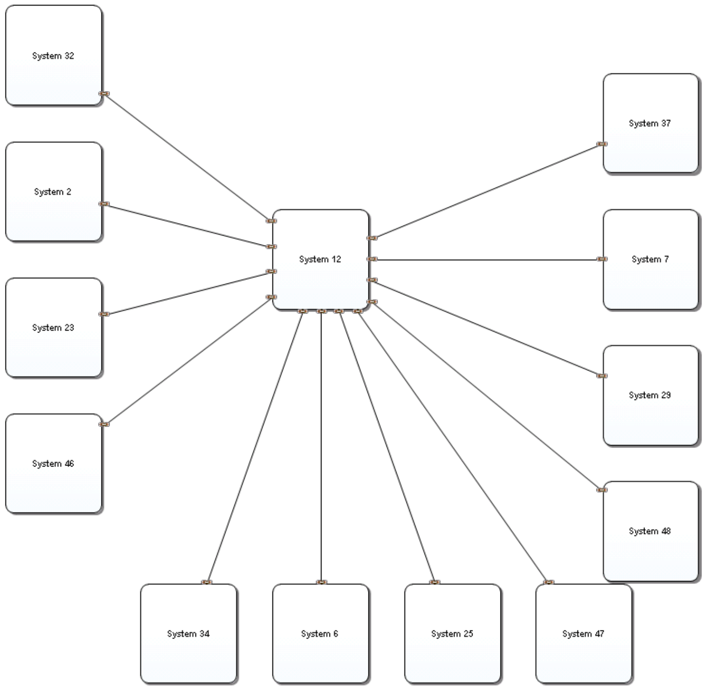


Figure 9.9.: *Embarcadero EA/Studio*- axis Impact Analysis and Reporting: Impact analysis diagram

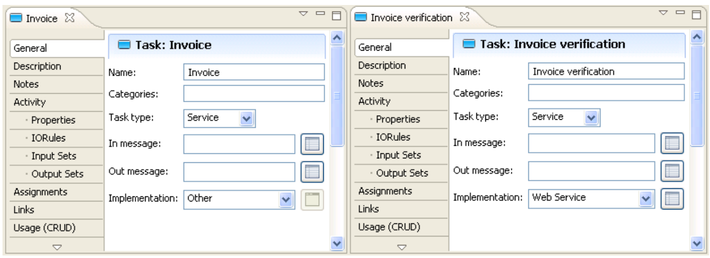


Figure 9.10.: *Embarcadero EA/Studio*- axis Usability: Comparing two objects' attributes

9.2. Statement to Evaluation of EA Management Support

For *Embarcadero EA/Studio* no evaluation of the scenarios concerning the EA management support have been simulated. This was decided in accord to Embarcadero Technologies, Inc., as the information model of the tool is fixed to a set of classes, attributes, and relationships (see Section 9.1.5), that cannot be considered sufficient for executing the EA management tasks as simulated in the scenarios as outlined in Section 4.2. As a consequence, no sensible evaluation results could be created. According to Embarcadero Technologies, Inc., it is planned to extend the predefined information model in future releases, which then would potentially introduce concepts as needed for supporting the EA management tasks as described in Section 4.2. If this was the case, a lot of the functionalities as described in Section 9.1 could proof to be useful in actually performing EA management tasks. Nevertheless, although no support in that area is currently provided, we have chosen to include the evaluation results for the scenarios of specific functionalities, in order to give an indication on the overall functionality provided by *Embarcadero EA/Studio*.

9.3. Tool Vendor's Profile

Dr. Sultan Shiffa

Overview

Embarcadero Technologies, Inc. delivers professional grade database tools that companies use to design, develop, and manage databases and the data they contain. More than 12,000 customers worldwide and more than 90 of the Fortune 100 rely on Embarcadero's cross-platform tools to reduce complexity, improve productivity, and strengthen security. The company's flagship database tools include: ER/Studio, DBArtisan, Rapid SQL, and Change Manager. Founded in 1993, Embarcadero Technologies is headquartered in San Francisco with offices in Melbourne, Australia; Munich, Germany; and Maidenhead, United Kingdom.

Established

October 1993

Key Differentiators

Cross-platform database tools for the entire database management lifecycle

1. Ease of use – quick to master
2. Enhanced productivity – fast ROI

Embarcadero Offerings

Cross-platform database tools for the entire database management lifecycle Embarcadero's professional grade database tools span the entire database lifecycle.

DATABASE DESIGN

Embarcadero's database design tools ensure that an organization will always know where its data lives, what it means, and how it should be used. A model-driven approach ensures that data architects and business users can collaborate effectively through clear documentation and common business definitions.

1. **ER/Studio®**– Design and model databases in an intuitive, visual environment
2. **Embarcadero EA/Studio™** – Enables business users and data architects to model business processes, then relate those processes to data
3. **Embarcadero Schema Examiner™** – Automated error-checking for database schemas

DATABASE DEVELOPMENT

Embarcadero's cross-platform database development tools raise developer productivity by providing consistent, intuitive interfaces across disparate database platforms. Our comprehensive tools simplify

SQL scripting, object management, version control, and schema deployment, enabling developers to create, edit, and optimize high-performance SQL.

1. **Rapid SQL**®– Write, debug, and tune complex database SQL more efficiently to produce high-performance, quality code in less time.
2. **Embarcadero Change Manager**™ – Automate schema capture, schema compare, and schema synchronization to track complex changes and minimize risks associated with change.

DATABASE MANAGEMENT

Embarcadero's cross-platform database management tools enable DBAs to adopt a consistent approach to database administration, performance, and security. By automating routine tasks, DBAs can proactively pinpoint and resolve issues before they become a threat to database availability.

1. **DBArtisan**®– Administer databases across platforms from a single console. Maximize database performance, availability, and manage-ability.
2. **Embarcadero Performance Center**™ – Ensure maximum database availability and performance with 24/7 server-side performance monitoring.
3. **Embarcadero DSAuditor**™ – Understand how your database is being used, who is using it, and what they are doing, so you can easily optimize performance and ensure regulatory compliance.

Customer Highlights

- More than 12,000 companies
- More than 90 of the Fortune 100
- More than 100,000 end users

CHAPTER 10

Hewlett Packard (HP Project and Portfolio Management Center)

Contents

Hewlett Packard provides the tool *HP Project and Portfolio Management Center*, which mainly focuses on the IT project and portfolio management process. The *HP Project and Portfolio Management Center* shall provide visibility and control over the demands, the portfolio of IT projects, and the roll-out of strategic changes.

Section 10.1 contains a statement, explaining why we did not analyze the *HP Project and Portfolio Management Center*, followed by a tool vendor's profile of *Hewlett Packard* in Section 10.2.

10.1. Statement to Analysis of HP Project and Portfolio Management Center

During the preparation phase of this survey, we identified different tools in the area of EA management, using the input of our industry partners and internet research activities. After the identification phase, we asked our industry partners to vote on the different tools. Thereby, the *HP Project and Portfolio Management Center* was ranked high by our industry partners (cf. table 1.1) as a tool of major interest.

Based on our experience made in 2005, we agreed with *Hewlett Packard* that a detailed analysis and evaluation would fail the intended usage of *HP Project and Portfolio Management Center.* The scenarios we built in cooperation with our project partners mostly cannot be simulated with the *HP Project and Portfolio Management Center* due to the following reasons:

HP Project and Portfolio Management Center concentrates on management of projects, portfolios, strategies, etc., but in the underlying information model associations between the business layer[1], consisting of business processes, organizational units, etc. and the application layer, consisting of application systems, components, etc. do not exist. Therefore, e. g. a business processes cannot be connected to business applications using a *support-relationship.*

HP Project and Portfolio Management Center does not focus on continually planning and developing the EA like demanded in the scenarios *Landscape Management* (see section 4.2.1), *Synchronization Management* (see section 4.2.4), *Strategies and Goals Management* (see section 4.2.5), *Business Object Management* (see section 4.2.6), *SOA Transformation* (see section 4.2.7, and *IT Architecture Management* (see section 4.2.8), which all cope with conceptional elements of the EA, like e. g. planned or target landscapes.

Hence, we agreed with *Hewlett Packard* that most of the scenarios concerning EA management support, see above, cannot be simulated using their product and therefore, we decided not to analyze *HP Project and Portfolio Management Center* within the scope of our survey.

The statements above do not refer to the capabilities of *HP Project and Portfolio Management Center* as a project portfolio management tool. These statements only reflect the capabilities as EA management tool, like introduced in section 3.

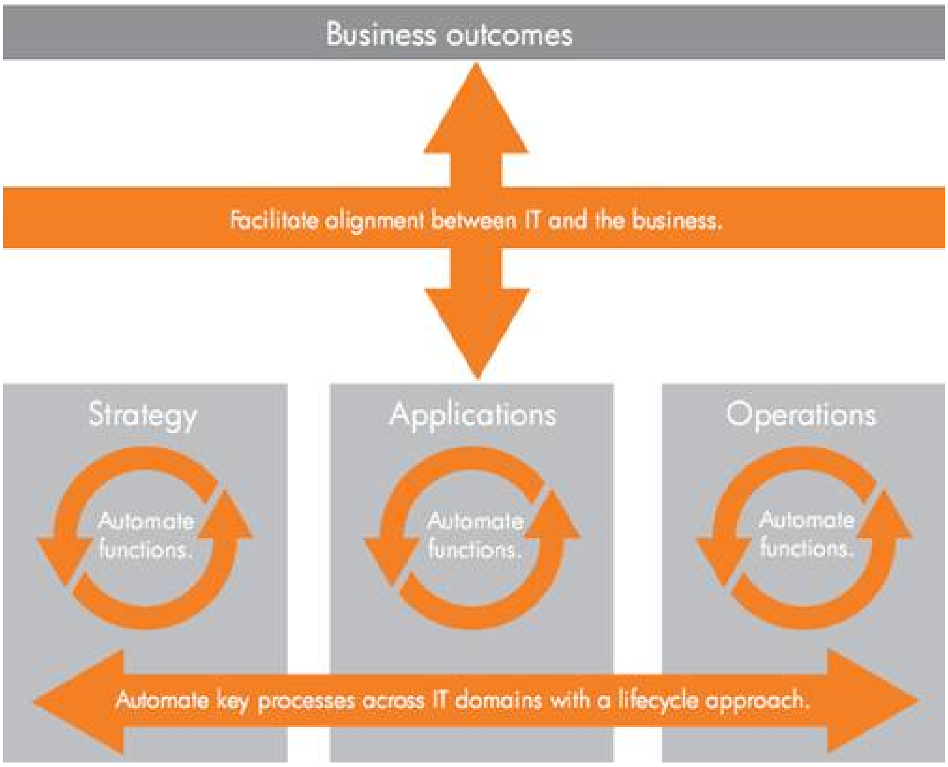
[1]For a deeper introduction to these layers of an information model see section 3.2.

10.2. Tool Vendor's Profile

Alex Osterloh, Presales PPM Germany, HP Software Group

HP BTO Software

HP BTO Software helps your organization bridge the gap between IT and the lines of business to keep your initiatives fully aligned with business goals and priorities. This enables you to verify that the business is getting what it wants from IT. HP BTO software also optimizes key functions within IT strategy (CIOs and CTOs), IT applications and IT operations, the typical IT domains. In addition, it automates key processes across functional IT silos. By doing so, HP BTO software enables different areas of IT to work together in a coordinated and collaborative way—rather than in silos—to accelerate and drive positive business outcomes.



HP Project and Portfolio Management Center

HP Project and Portfolio Management Solution reside in the Strategy pillar. HP Project and Portfolio Management Center software provides your PMO with visibility into strategic and operational demand as well as in-flight projects and programs. HP Project and Portfolio Management Center:

- Offers top-down planning capabilities that are supported with bottom-up detailed project plans
- Helps your IT organization quickly adapt to unforeseen changes and still maintain alignment with business objectives
- Enables execution of work in a manner that delivers business results
- Enhances visibility and enable controls to maintain compliance

HP Project and Portfolio Management Center components

HP Portfolio Management software enables you to govern your portfolio of IT projects, applications and opportunities in real time with effective collaborative processes. It gives you the information and process to make effective portfolio decisions—from proposal initiation, justification and review to project initiation, execution, deployment and benefits realization. And with the optional Portfolio Optimization feature, HP Portfolio Management automatically determines the best mix of proposed projects, projects and assets based on user-defined criteria.

HP Portfolio Management software enables you to govern your portfolio of IT projects, applications and opportunities in real time with effective collaborative processes. It gives you the information and process to make effective portfolio decisions—from proposal initiation, justification and review to project initiation, execution, deployment and benefits realization. And with the optional Portfolio Optimization feature, HP Portfolio Management automatically determines the best mix of proposed projects, projects and assets based on user-defined criteria.

HP Program Management software enables you to collaboratively manage your programs from concept to completion. It automates processes for managing scope, risk, quality, issues and schedules. With HP Program Management, you no longer need multiple point tools and paper manuals to manage program initiation, approval, scope changes, risk, issue resolution, resources or status.

HP Project Management software helps you meet the challenges of managing projects in large, geographically dispersed enterprise environments. It integrates project management and process controls to reduce the number of project/schedule overruns, thereby reducing project risks and costs.

HP Financial Management software provides real-time visibility into all financial attributes related to programs, projects and overall IT demand—without costly integrations to multiple data sources. For global organizations, multi-currency is supported within the application. HP Financial Management supports comprehensive earned value analysis and SOP 98-1 capitalization.

HP Resource Management software provides comprehensive resource analysis, which includes both strategic and operational activities at any stage in the work lifecycle. This holistic approach enables a complete understanding of where resources are currently committed and allocated. In turn, your managers can quickly respond to changes with a clear understanding of the effects on resource capacity and work prioritization.

HP Time Management software helps you focus on value-added activities by streamlining time collection and improving accuracy across the wide range of work performed by IT. This provides the capabilities your IT organization needs to overcome the challenges of time collection and to focus on improving productivity and execution.

HP Demand Management software captures all requests on IT so you'll know what the organization is asking for and have the information to focus your valuable IT resources on top business priorities. Stakeholders have a comprehensive picture of past, present and future demand so requests can be prioritized, assigned, viewed and "sliced-and-diced" across multiple dimensions to identify trends.

HP Project and Portfolio Management Dashboard provides role-based, exception-oriented visibility into IT trends, status and deliverables to help you make and execute real-time decisions. It supports information sharing with other applications or corporate portals via enterprise industry standards JSR 168 and WSRP.

HP Project and Portfolio Management Foundation is our unique platform that runs the application. It includes our advanced workflow engine and configuration capabilities. Additionally, Project and Portfolio Management Foundation provides enterprise-class data security features.

CHAPTER 11

IBM (Rational Software Architect)

Contents

IBM provides the tool *Rational Software Architect*, which mainly focuses on the model-driven developement with the UML for creating well-architectured applications and services. The *Rational Software Architect* shall provide capabilities for Eclipse-based UML Modeling and Java development.

Section 11.1 contains a statement, why we did not analyze the *Rational Software Architect*, followed by a tool vendor's profile of *IBM* in Section 11.2.

11.1. Statement to Analysis of IBM Rational Software Architect

During the preparation phase of this survey, we identified different tools in the area of EA management, using the input of our industry partners and internet research activities. After the identification phase, we asked our industry partners to vote on the different tools. Thereby, the *Rational Software Architect* was ranked high by our industry partners (cf. Table 1.1) as a tool of major interest.

In contrast to *Rational Software Architect* being perceived as a tool suitable for EA management by some of our industry partners, our initial analysis of the tool showed, that most of the scenarios from Section 4 cannot be executed with the *Rational Software Architect*. We therefore agreed with *IBM* not ot perfom a full-scale evaluation of the *Rational Software Architect*, but to provide a set of reasons, why the tool cannot be used for our scenarios.

Additionally, we would like to emphasize, that the statements below do not refer to the capabilities of *Rational Software Architect* as a object-oriented modelling tool or a tool for model-driven developement.

The *Rational Software Architect* concentrates on modeling and design aspects of single software systems. Nevertheless, the *Rational Software Architect* can be used to design and model the various elements contained in the different layers of the EA, as well as the relationships between the layers[1]. Thereby, the flexibility of the created models is based on and limited to the posibilities defined within the UML.

Concerning the scenarios for analyzing specific functionality, as introduced in Section 4.1, the *Rational Software Architect* does not provide process guidance or role-based views. Thereby, especially the scenarios dealing with multiple user support and process guidance like required in the scenarios *Supporting multiple Users and collaborative Work* (see Section 4.1.9) and *Supporting lightweight Access* (see Section 4.1.5) cannot be simulated with the *Rational Software Architect*.

The *Rational Software Architect* does not focus on continually planning and developing the EA like demanded in the scenarios *Landscape Management* (see section 4.2.1), *synchronization management* (see section 4.2.4), *Strategies and Goals Management* (see section 4.2.5), *Business Object Management* (see section 4.2.6), *SOA Transformation* (see section 4.2.7, and *IT Architecture Management* (see section 4.2.8), which all cope with conceptional elements of the EA, like e. g. planned or target landscapes. The time dimensions could only be simulated through the maintenance of different models, thus accepting the drawback, that the relations between different time-versions of the same object are lost.

[1]For a deeper introduction to these layers of an information model see section 3.2.

11.2. Tool Vendor's Profile

Daniel Heckmann, Technical Sales IBM Deutschland

IBM Rational Software helps organizations automate, integrate, and govern the core business process of software and systems delivery via the *IBM Rational Software Delivery Platform*. Offering an array of products, services, and best practices, this open, modular, and proven solution spans the entire software and systems delivery lifecycle.

The *IBM Rational Software Delivery Platform* comprises products in four lifecycle categories and provides the basis for a set of targeted solutions:

- Architecture management: Model, design, and rapidly build resilient architectures for SOAs, systems, and applications.
- Change and release management: Improve software delivery and lifecycle traceability, from requirements through deployment.
- Process and portfolio management: Align business goals, best practices, and projects for improved productivity and predictability.
- Quality management: Ensure software functionality, reliability, and performance throughout development and production.
- Targeted solutions: Solve lifecycle management & governance challenges for industries, SOA, systems & more.

IBM Rational and Enterprise Architecture Management

Deliver value from inception to implementation

Enterprise architecture (EA), the "blueprint" for systematically and completely defining an organization's current (baseline) or desired (target) environment, coupled with a process for development and maintenance is becoming widely accepted as a mechanism for enterprise transformation. As a key planning discipline, EA helps guide and optimize an organization's IT investments and translate business strategies into technology solutions. It bridges the gap between systems and application discovery, development and deployment – linking IT activities to the business mission, integration and systems modernization efforts, and ensures deployment of resources in the most efficient manner.

The *IBM Rational Software Delivery Platform* helps operationalize enterprise architecture program success through these best practices:

- Balance portfolios and investments, align resources with business case objectives
- Ensure requirements drive your architecture
- Accelerate application development with integrated line of business process and application models
- Automate traceability from inception to implementation
- Leverage service oriented design & development processes and techniques to achieve reuse and reduce risk
- Control change as your architecture evolves
- Protect investments by encouraging reuse

Design your enterprise applications - Rational Software

Rational Software Architect offers a powerful, integrated design and construction environment that helps software architects understand, design, manage and evolve enterprise solutions and services across the team, across the world and across different areas of technical expertise. Part of a flexible family of business driven development products that are built on the open Eclipse integration platform, Rational Software Architect offers extended support for the Object Management Groups (OMGs) industry standard Unified Modeling Language (UML). And the software's many powerful visual modeling and editing features are designed to improve productivity, enhance architectural control and ease the design-to-code experience for Java and Java 2 Enterprise Edition (Java EE), Web services, service oriented architecture (SOA) and C/C++ applications.

Model applications more productively than ever with UML 2

Rational Software Architect, based on UML, Version 2.1, includes new ease-of-adoption and ease-of-use features that raise the bar for user productivity in modeling workflows. Its support for a variety of diagram types, including freeform diagram support, aids in design, discovery and documentation activities.

Exploit the latest in modeling-language standards

Rational Software Architect supports the OMGs Model Driven Architecture (MDA) initiative by allowing the user to define multiple levels of models coupled with user-defined transformations between models and code, resulting in a clearer separation of concerns across the lifecycle.

Tap into the power of model-based automations

Modeling alone adds value and helps reduce project risk, but even greater benefits are realized when models are used to automate the creation of other development artifacts, including other models, code and more. Use the design patterns included with *Rational Software Architect* to help you more quickly build the content of your UML analysis and design models. Or, create your own UML-based patterns to extend this benefit even further. *Rational Software Architect* supports model-to-code and code-to-model transformations.

Modeling in Enterprise Architecture Management

Because of it's high flexibility, the extensibility through UML profiles and custom model to text and model to model transformations, *Rational Software Architect* is an ideal tool to design your enterprise applications and describe the landscape they are residing in. The reuse of models, generalization to model templates and the generation of other artifacts and models out of them speeds up SW development. Using UML models in *Rational Software Architect* it is possible to visually describe the aspects of the IT environment and systems and reuse the models in other projects. This avoids inventing the wheel over and over again. Using the tight integration with IBM Rational Data Modeler, this goes from the business process description with use cases and activity diagrams, through the logical and physical data models, service modeling, component modeling and the design model down to modeling the infrastructure.

Rational Software Architect integrates completely into the *IBM Rational Software Delivery Platform*. This allows importing business processes from Websphere Business Modeler, generating a service model from the process descriptions and generating service implementations from the model. The integration to Rational's requirements management software *RequisitePro* enables customers to connect the elements in models to the relating requirements and their written description. This creates transparency and shows why a certain element was put in the architecture and what the impact is if it is omitted.

Rational Software Architect is a complete customizable modeling environment. Using self defined meta models architects can describe the architecture in terms relevant to their company, which gives customers a high flexibility in what they describe about their IT instead of being dependent on a vendor provided meta model.

CHAPTER 12

IDS Scheer AG (ARIS IT Architect)

Contents

IDS Scheer AG has a long time background in the Business Process Modeling market, therefore, approaching the topic of EA management from this point of view. The tool evaluated below is the ARIS IT Architect 7.0.2 (Build 7.0.2.244618), which can be seen as integrated into *ARIS Design Platform*.

12.1. Evaluation of Specific Functionality

Below the results of simulating the scenarios from Section 4.1 with the *ARIS IT Architect* are explicated. Prior to the detailed description, the overall terminology of the tool is shorthanded and related to the terminology used throughout the rest of the survey. Concerning the information model used, the *ARIS IT Architect* relies on three different concepts:

- the *Objecttype*, which corresponds to a *Class* in our information model,
- the *Attributetype*, which corresponds to an *Attribute* in our information model, and
- the *Connectiontype*, which corresponds to an *Association* in our information model.

Each of these types is represented in the repository by a unique identifier, which can subsequently be used for mapping externally defined types, as e. g. classes or attributes, to the predefined types of the *ARIS IT Architect*.

Complementing the information model concepts, the *ARIS IT Architect* also supports the concept of the *viewpoint* on the information, as e. g. backing the visualizations used throughout the scenarios from Section 4.2. The viewpoints of *ARIS IT Architect* are called *Modeltypes*.

12.1.1. Importing, Editing, and Validating

The *ARIS IT Architect* provides access to a multitude of import formats, among others XML (via transformation to the ARIS specific dialect of *AML*), XMI[1], CADM[2], and Microsoft Excel. Subsequently the import of data from Microsoft Excel is further detailed.

For importing data about the enterprise architecture the *ARIS IT Architect* provides a workbook of standardized structure, of which an empty template instance can be created using a reporting function from the *ARIS IT Architect*. After the data has been restructured to fit the structure of the import template, a set of basic transformations, which should be performed during the import, can be specified. These transformations target the mapping between the classes used in the import data and the objecttypes from the *ARIS* methodology. Similarly, mapping relations between the input attributes and the attributetypes as well as between the input relationships and the connectiontypes can be defined. More sophisticated transformations, as e. g. splitting comma separated values maintained in a single spreadsheet column, are not directly supported by the importing mechanism, but can easily be realized using the built-in functionalities of Microsoft Excel.

The extensive *ARIS* method documentation is very helpful during the import, especially in determining the mapping between concepts, which should be imported. The documentation (user guide) is also helpful, in finding the appropriate methods, objecttypes, and reports provided by the tool for performing the different tasks related to EA management. Nevertheless, the advice of a consultant with experience in the methodology is deemed advisable. Having had advice of that kind, most of the concepts of the information model of the survey could be mapped to concepts from the *ARIS* methodology, while especially concerning attributes, the full extent of options to adapt the predefined

[1] *XMI* is an model interchange format, developed by the Object Management Group (OMG).
[2] *CADM* is the Core Archtecture Data Model used in model exchange in the US Department of Defense.

metamodel had to be utilized. Additionally, some concepts, as e. g. the `Demand` had no immediate counterpart in the predefined information model, such that an alternative solution was advised by the consultant.

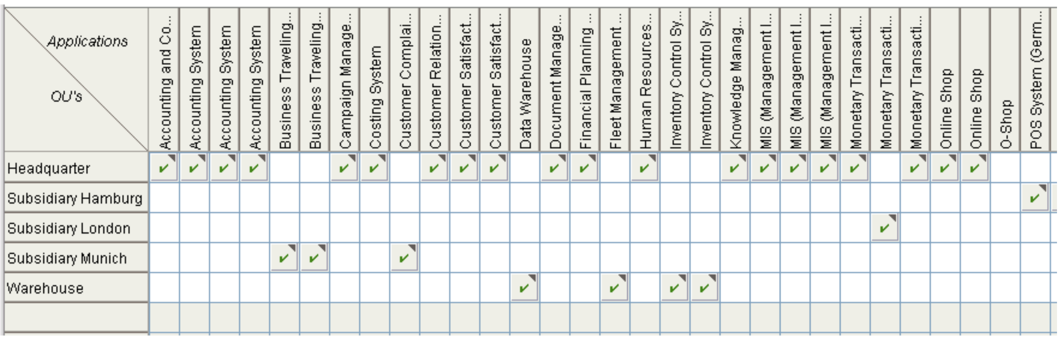
Concerning the validation of imported data, the *ARIS IT Architect* can provide such validation information in a Microsoft Word report as shown in Figure 12.1. Therein, incorrect values in the imported file are reported on attribute and connection level - to the extent, the information imported does not conform to the (mapped) information model of the *ARIS IT Architect*. Consequently, the metamodeling capabilities of the *ARIS IT Architect* strongly influence the validation capabilities. For a detailed evaluation of the metamodeling capabilities of the *ARIS IT Architect* see Section 12.1.4.

Page	Row/Column	Error
Objects	2 / 5	Attribute 'maintenanceCosts' is not valid for this object/connection.
Objects	2 / 6	Attribute 'version' is not valid for this object/connection.
Objects	2 / 9	Unable to assign value '38869.0' to attribute 'plannedFrom'.
Objects	2 / 10	Unable to assign value '38990.0' to attribute 'plannedTo'.
Objects	2 / 11	Unable to assign value '38991.0' to attribute 'developmentFrom'.
Objects	2 / 12	Unable to assign value '39263.0' to attribute 'developmentTo'.
Objects	2 / 13	Unable to assign value '39264.0' to attribute 'productionFrom'.
Objects	2 / 14	Unable to assign value '39461.0' to attribute 'productionTo'.
Objects	2 / 15	Unable to assign value '39462.0' to attribute 'retirementFrom'.
Objects	2 / 16	Unable to assign value '39782.0' to attribute 'retirementTo'.
Cxns	2 /	Source object '110001.0' does not exist.
Cxns	3 /	Source object '110001.0' does not exist.
Cxns	4 /	Source object '110005.0' does not exist.
Cxns	5 /	Source object '110005.0' does not exist.
Cxns	6 /	Source object '110001.0' does not exist.
Cxns	7 /	Source object '110004.0' does not exist.
Cxns	8 /	Source object '110005.0' does not exist.
The import was completed successfully.		

Figure 12.1.: *ARIS IT Architect* - axis Importing, Editing, and Validating: Report showing validation result during data import

The user is provided convenient access to the data stored in the repository of the *ARIS IT Architect* via the user interface. A tree-like view enables easy navigation in the model, while a table-like editor exists, in which the attribute values for a single instance or a list of instances can be displayed. Via this editor convenient editing of mass data is directly supported within the user interface of the *ARIS IT Architect*, e. g. by copying or moving values. In order to leverage easy-to-use edit mechanisms for relationships a so called *matrix-editor* is provided (cf. Figure 12.2). For this editor, the user can select two objecttypes and a corresponding connectiontype. The objects of the selected types are subsequently displayed in the editor as rows and columns respectively, while existing connections between objects are represented by a clickable symbol in the intersection of the corresponding rows and columns. Via this interface the user can create new connections and delete existing ones, if the connection to be deleted is not used in any model.

If one would like to use an external editor for changing information from the repository, the user can export the data to e. g. an Microsoft Excel file and perform changes there. Subsequently, the data can be reimported into the *ARIS IT Architect*, performing the validation as described above. No direct conflict resolution mechanisms are provided during the reimporting, as no edit-time information is stored in the Microsoft Excel export. Nevertheless, conflicting changes can be prevented making use of locking data as described in Section 12.1.5. According to IDS Scheer, data changes in other tools should be performed utilizing the XML based exchange mechanisms, which also provides the facility to choose, whether to overwrite the elements in the repository or not. In addition to that,



Applications / OU's	Accounting and Co...	Accounting System	Accounting System	Accounting System	Business Traveling...	Business Traveling...	Campaign Manage...	Costing System	Customer Complai...	Customer Relation...	Customer Satisfact...	Customer Satisfact...	Data Warehouse	Document Manage...	Financial Planning ...	Fleet Management ...	Human Resources...	Inventory Control Sy...	Inventory Control Sy...	Knowledge Manag...	MIS (Management I...	MIS (Management I...	MIS (Management I...	Monetary Transacti...	Monetary Transacti...	Monetary Transacti...	Online Shop	Online Shop	O-Shop	POS System (Germ...
Headquarter	✓	✓	✓	✓			✓	✓		✓	✓	✓		✓	✓		✓			✓	✓	✓	✓	✓		✓	✓	✓		
Subsidiary Hamburg																														✓
Subsidiary London																									✓					
Subsidiary Munich					✓	✓			✓																					
Warehouse													✓			✓		✓	✓											

Figure 12.2.: *ARIS IT Architect* - axis Importing, Editing, and Validating: The matrix editor for associations

the maintenance of application related information is supported via the inventory of the *ARIS IT inventory*[3].

Rating: 5

12.1.2. Creating Visualizations

The *ARIS IT Architect* provides a multitude of different modeltypes for graphically displaying distinct objecttypes and connectiontypes from the information model. These modeltypes affect the visualization capabilities to the extent, that specific types of objects or connections can only be visualized in specific modeltypes, which are according to the underlying methodology of the *ARIS IT Architect* used in different scenarios. These usage scenarios are explained in detail in the supplied method documentation. Consequently, the actual visual models modeled in accordance to the *ARIS* modeltypes are consistent with the intended modeling capabilities.

Taking into account the overall visualization capabilities provided by the *ARIS IT Architect*, it has to be noted, that most modeltypes can be seen as *graphlayout maps* in terms of our definition (cf. Chapter C). These visualizations can be automatically generated, i. e. the user can select objects from an appropriate objecttype to be visualized in the model. In a second step, the user can chose a connectiontype between these objecttypes, which should be displayed in the visualization, leading to a generated visual model of a graphlayout buildup as shown in Figure 12.3.

Visualizations conforming to the other software map types as described in the Section 4.1.2 can also be created, but employ more manual interaction of the user. Additionally, the semantic conformance between the model and the underlying information is thus not always automatically maintained, especially concerning the *cluster map* (see Figure 12.4) and the *process support map* (see Figure 12.5). There, the relative position of the symbols, e. g. the alignment of an application system symbol to a business process symbol, does not have a well-defined semantics - only for selected objecttypes positioning is performed in a semantically consistent manner. Nevertheless, it has to be noted, that concerning the process support map, an exception exists: it is possible to automatically create a process support map displaying, which application system supports which business process (represented as function) at which organizational unit. An example of such a visualization is shown in Figure 12.5.

Concerning the *time interval map*, the situation presents somehow different. Visualizations of that type can automatically be created as reports (called *lifecycle reports* in the tool), e. g. from lifecycle information present at application systems (see Figure 12.6). Being reports and not models in the *ARIS IT Architect*, time interval maps cannot be changed or edited manually, furthermore ensuring

[3]According to IDS Scheer, the *ARIS IT inventory solution* is going to be released as a product in Q2/2008.

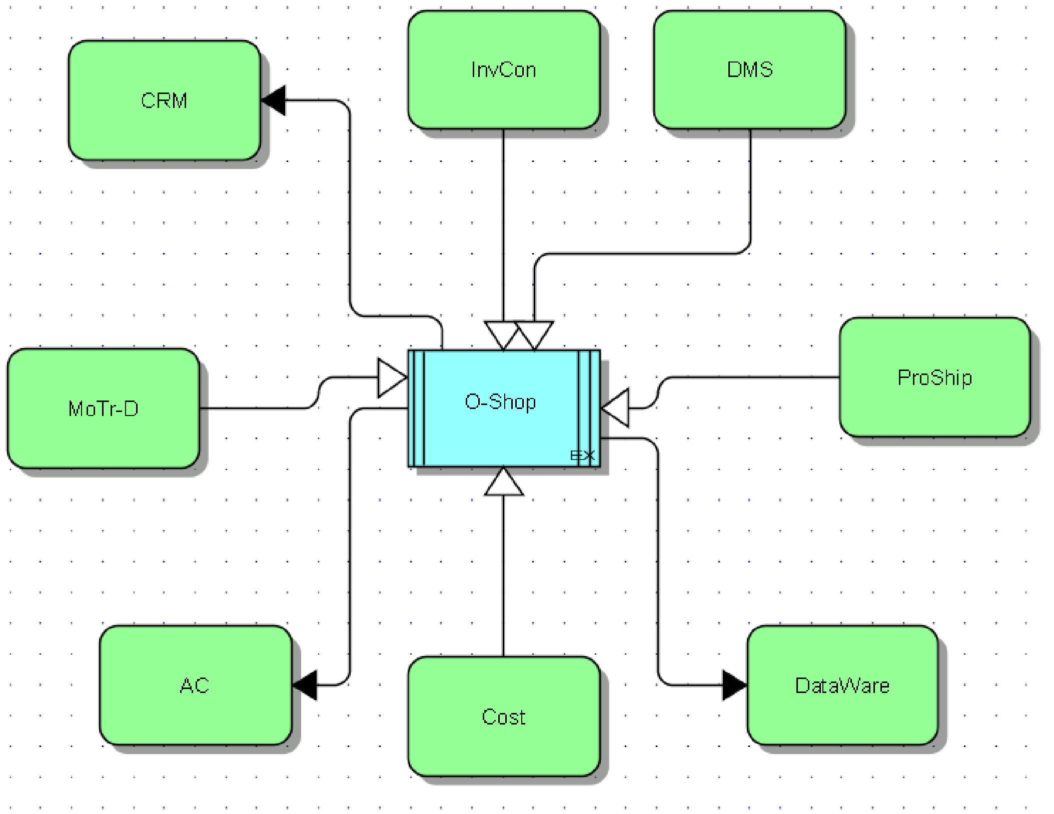


Figure 12.3.: *ARIS IT Architect* - axis Creating Visualizations: Graphlayout map

consistency between the visualization and the information visualized. Nevertheless, this type of report is out-of-the-box dedicated to present lifecycle information, if other temporal information should be visualized similarly, adaptations to a configuration file of the *ARIS IT Architect* had to be applied[4].

A *swimlane diagram*, as described in Section 4.1.2 cannot automatically be created, but can be layouted employing some manual interaction, leading to a visualization as in Figure 12.7.

The creation of portfolio matrix diagrams is partially supported by the *ARIS IT Architect*, via a specialized report, called *System evaluation report*. In this report, as suported out-of-the-box, the user can select a number of application system types and place them in a portfolio matrix according to their crisiticality and a specific cost attribute (cf. Figure 12.8). If portfolio matrices showing other classes from the information model or indicating different attributes should be created, a configuration file in the *ARIS IT Architect* could be adapted appropriately. Furthermore, the user could also create portfolio matrices relying on Microsoft Excel, which can conveniently operate on exported data. Thereby, it would also be possible to automatically adapt the size of the symbols for presenting information on the value of a certain attribute.

Rating: 5

[4]According to *IDS Scheer*, a future release of the tool will expose this configuration facility directly in the user interface.

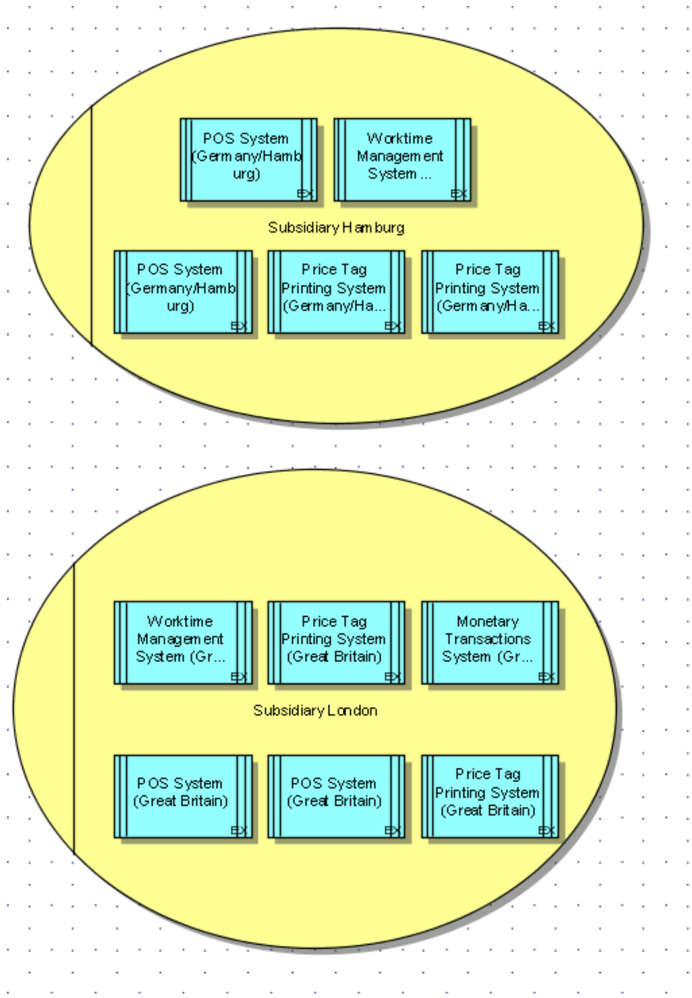


Figure 12.4.: *ARIS IT Architect* - axis Creating Visualizations: Manually created cluster map

12.1.3. Interacting with, Editing of, and Annotating Visualizations

The *ARIS IT Architect* provides various ways of editing an existing visualization of underlying information within the user interface. Some of these functionalities therein target the visual makeup of the models, while others also have semantic implications, i. e. change data in the underlying repository. The first group of interaction paradigms include resizing and moving symbols as well as rerouting lines representing a connection without changing the actual endpoints of it. Also, the user can choose for every instance of an objecttype the actual symbol for representation from a set of predefined symbol templates for that objecttype. Consequently, the chosen symbol is then used for every occurence of the corresponding object in every diagram.

When concerning the possibility to e. g. edit portfolio matrices or time interval maps, a limitation to the interaction capabilities applies, as these visualizations are no models in the terminology of the *ARIS IT Architect*, but are reports, which cannot be edited in the tool, once they have been created.

In cluster map like visualizations, moving a symbol can have an additional semantic implication, occuring, when a symbol is nested into another one. Then, the *ARIS IT Architect* displays a dialog, in which the user is asked, which connectiontype is expressed by this nesting and should thus e. g.

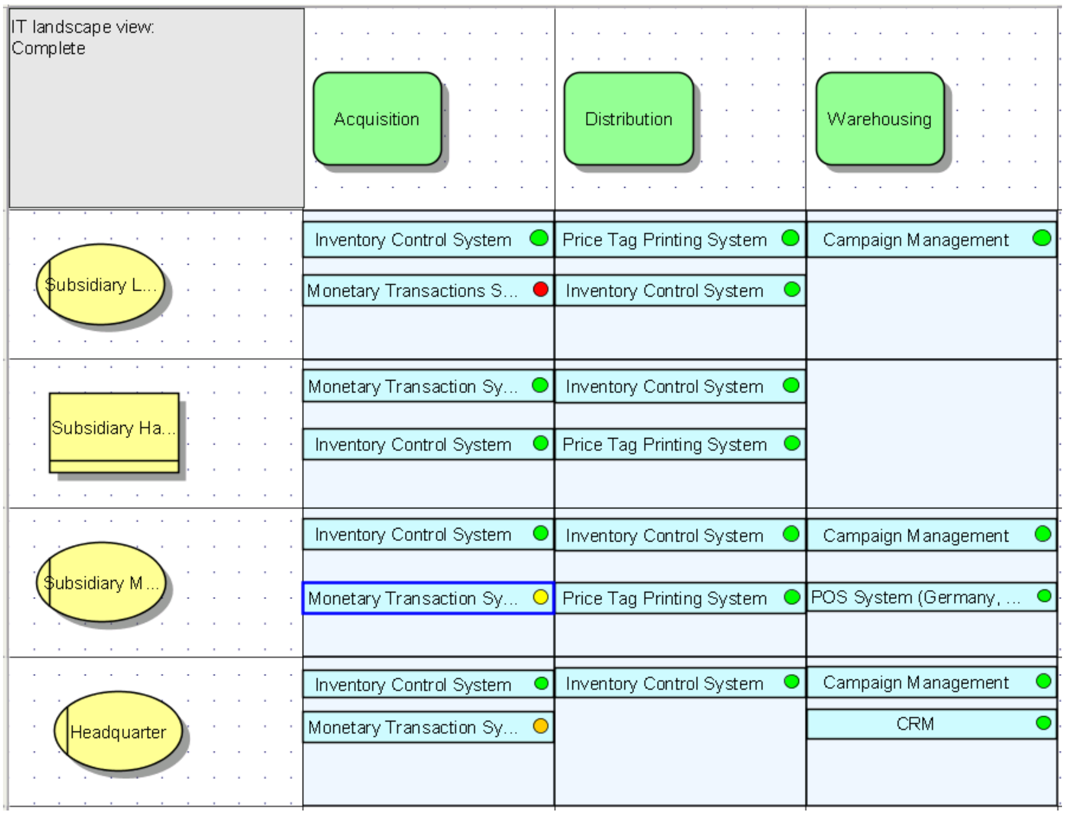


Figure 12.5.: *ARIS IT Architect* - axis Creating Visualizations: Semi-automatically create process support map

Figure 12.6.: *ARIS IT Architect* - axis Creating Visualizations: Time interval map

be created. An association displayed by a nesting of symbols is consequently maintained, even if the inner symbol is dragged out of the other one - visually representing the connection by an automatically created line.

Changing a visual variable of a symbol, e. g. its color (cf. Figure 12.9) in accordance to the value of an attribute of the represented object, is not directly supported by the *ARIS IT Architect*, but could be leveraged by customizing the *Lifecycle Management Report* as shipped with the tool. Nevertheless, consistency between the visualizations and the actual data represented is partially enforced, as removing symbols representing objects or connections does not affect the underlying data, if the same entity from the repository is visualized on another diagram. This mechanism prevents the data from acciden-

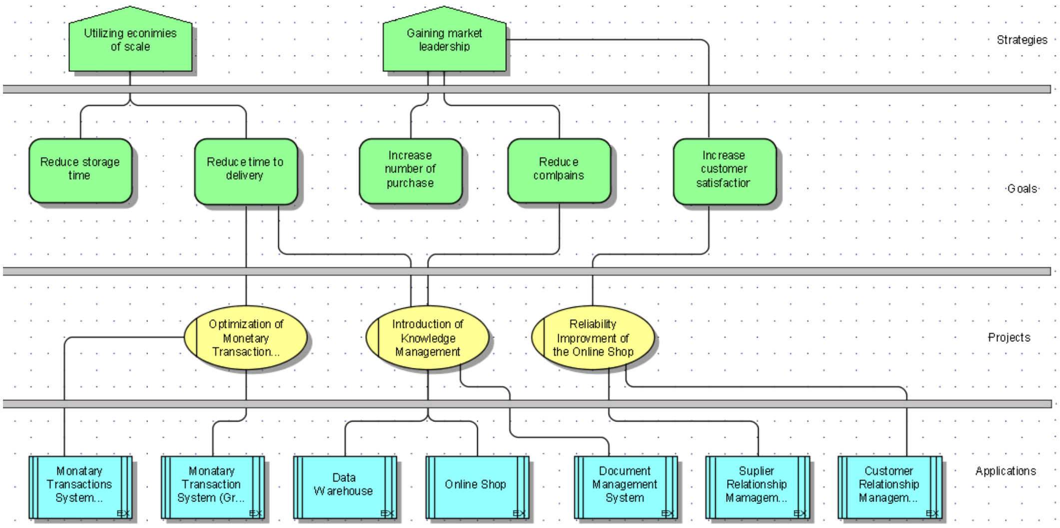


Figure 12.7.: *ARIS IT Architect* - axis Creating Visualizations: Swimlane diagram

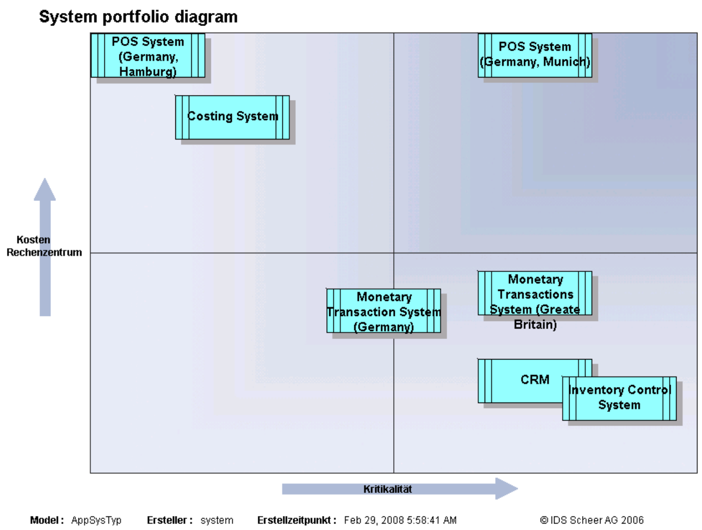


Figure 12.8.: *ARIS IT Architect* - axis Creating Visualizations: Portfolio matrix report

tal changes, therefore enhancing the consistency between visualization and data, although sometimes a shorthanded way for both deleting the symbol and represented element would be desirable. The latter can be achieved in a batch-processing manner by utilizing the administrative tools for executing a repository reorganization.

Rating: 5

12.1.4. Flexibility of the Information Model

The approach of the *ARIS* methodology relies on a comprehensive predefined information model, which is shipped with the *ARIS IT Architect*. Consequently, the possibilities to adapt the information model are limited in some ways, e. g. no part of the information model, i. e. neither an objecttype, an

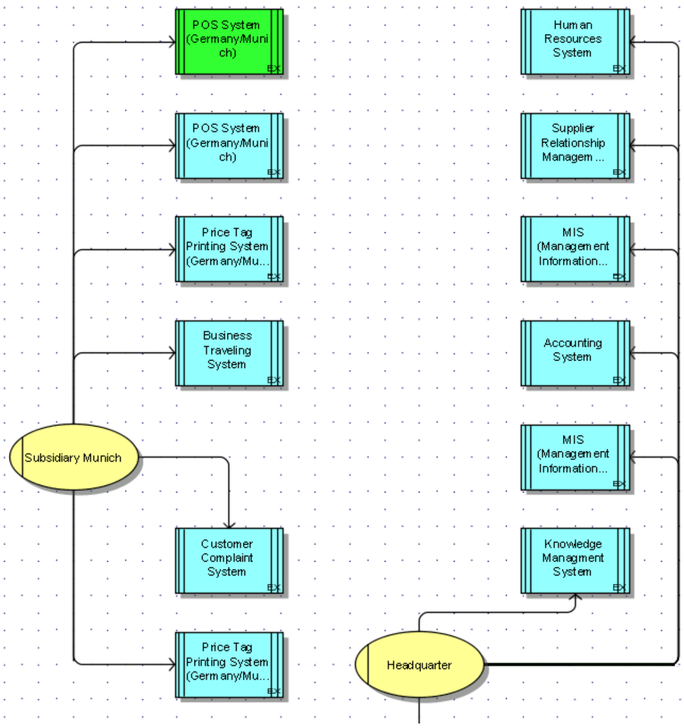


Figure 12.9.: *ARIS IT Architect* - axis Interacting with, Editing of, and Annotating Visualizations: Annotated graphlayout map

attributetype, nor a connectiontype can be deleted. Nevertheless, each concept can be hidden using the filtering mechanisms provided by the tool.[5] Replacing the capability to delete concepts by the option to hide them, additionally, removes the need for mechanims, which are needed to ensure model consistency in these cases. In order to provide further adaptability of the information model, all the concepts, as e. g. attributetypes, objecttypes, and connectiontypes can be renamed.

The introduction of new attributetypes can be performed easily, as the predefined information model provides a number of *placeholder* attributetypes of different datatypes, shared among all objecttypes. These attributetypes, again, can be renamed or hidden, with the latter functionality being important, as all objecttypes share these attributetypes. The missing capability to introduce new connectiontypes is compensated by the ability to assign attributetypes to connectiontypes[6], therefore actually making it possible to *simulate* different variants of one connectiontype. In addition to that, as alluded to above, connectiontypes can be renamed, if they are not used in the original meaning or if the terminology is not exactly the wished one.

The information model as adapted by the user can be displayed in three tabular reports, a first one containing the objecttypes (for an example see Figure 12.10), a second one containing the attributetypes, and a third one containing the existing connectiontypes. Complementing the reports with graphical notations of the current information model, as e. g. UML class diagrams, is not directly supported in the *ARIS IT Architect*. Nevertheless, the user can export comprehensive information about the modeltypes, objecttypes, connectiontypes available to his role utilizing an XLS export of his configuration.

[5]This filtering of the information model is user or role specific, so a user specific cutout of the information model can be made accessible.
[6]This is similar to the concept of the *association class* from the *unified modeling language* (UML).

Name	Type number	API name
Action	284	OT_ACTION
Actor	97	OT_ACTOR
Application system	64	OT_APPL_SYS
Application system class	7	OT_APPL_SYS_CLS
Application system type	6	OT_APPL_SYS_TYPE
Architecture element	343	OT_ARCH_ELEMENT
Business object	150	OT_BUSY_OBJ
Business rule	360	OT_BUSINESS_RULE
Business segment	302	OT_BUSINESS_SEGMENT
Collaboration	286	OT_COLLABORATION
Collaboration instance ...	291	OT_COLLAB_INST_SET
Component	188	OT_CMP
Component instance	290	OT_COMP_INST

Figure 12.10.: *ARIS IT Architect* - axis Flexibility of the Information Model: Excerpt from the list of supported objecttypes

The concept of a predefined symbolic representation for a specific objecttype is extensively used in the *ARIS IT Architect*, as it is a core part of the underlying *ARIS* methodology. As the underlying information model of the *ARIS IT Architect* is directly accessable in the user interface, no standardized exporting mechanisms regarding the information model are provided. Nevertheless, on a database level the schema information can be accessed.

Rating: 3

12.1.5. Communication and Collaboration Support

In order to support lightweight access to the information gathered in the *ARIS IT Architect*, the *ARIS Design Platform* encompasses the *ARIS Business Publisher*. This tool allows to access the information as well as the diagrams stored within the repository of the platform using a browser. Therein, especially the dynamic capabilities have to be noted, as the information is not displayed in a static HTML page, but is dynamically retrieved from the underlying data store. Additionally, the access via the *ARIS Business Publisher* can be restricted using role-based access control and role-based views on the information. Furthermore, the makeup of the user interface can be customized to incorporate corporate design rules. Nevertheless, the capabilities of the web publisher are limited to mere reading access, while editing information is not directly supported[7]

Concerning role-based access, it is possible to define the visibility of objecttypes, attributetypes, and connectiontypes, thereby creating role specific filterings of the information displayed can be created. Mechanisms for notifying users about changes on objects are not directly supported within the *ARIS Design Platform* as staging mechanisms are not. Nevertheless, similar mechanisms can be achieved using different repositories, between which the information is synchronized during staging.

In order to avoid conflicts from multiple editing of the data in the repository, the *ARIS IT Architect* provides an implicit locking mechanism. Models are therein locked exclusively for single user editing, when opened; the same locking is given for attributes having undergone a change of value. The objects are contrastingly locked optimistically. In several cases, especially when concerning models displaying certain connections from the underlying repository, also a lock preventing the deletion of that connection is automatically established. The locks are subsequently released, when saving the changes to the repository.

[7] According to IDS Scheer, editing access will be granted in the *ARIS Business Publisher* via the *ARIS IT inventory solution* add-on, which is going to be released as a product in Q2/2008.

According to IDS Scheer, an additional solution for *Release Cycle Management (RCM)* is available, which can be used for creating and navigating the version history of information objects. This solution also provide capabilities for staging mechanisms based on different repositories for modelling, reviewing, releasing, and archiving. The *ARIS* RCM solution has not been evaluated in the tool survey, as it is currently not part of the *ARIS IT Architect*. According to IDS Scheer it is planned to include the versioning and history functionality in the upcoming release 7.1.

Rating: 5

12.1.6. Support of large scale Data

The *ARIS IT Architect* deals well with large scale data concerning the application landscape, containing numerous objects, especially when the import of a large data is regarded. Although the data import has no "live performance", it is executed considerably fast. In addition to this, the execution time of most actions, like e. g. creating and modifying attribute values is not visibly affected by the number of objects contained in the repository.

Contrastingly, the automated creation of models displaying a large number of objects and connections takes considerably longer as also the editing of connections via the matrix editor does. Nevertheless, the overall performance of operations changing the graphical makeup of a model, e. g. by moving a symbol does not change noticeably. For a cluster map containing a small number of objects, e. g. up to 50 organization units and 500 application system, no slowdown of the creation process can be perceived. When the creation of a larger model (for a cutout see Figure 12.11) containing 250 organizational units and 5000 application system is concerned, no real-time execution is possible on a common hardware environment. Nevertheless, once a clustermap is generated and saved into the repositority, it can be reopened as fast as other every map.

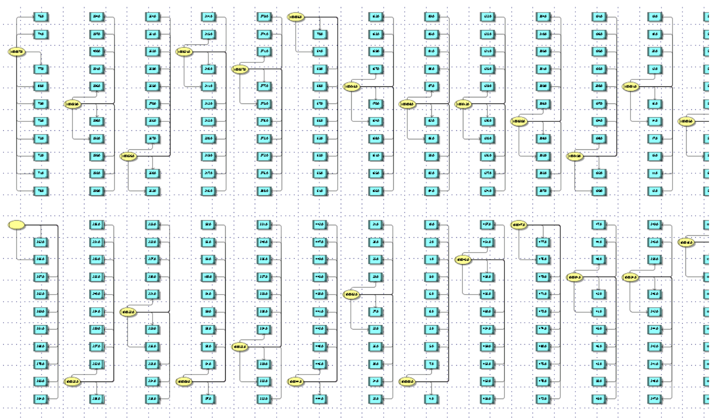


Figure 12.11.: *ARIS IT Architect* - axis Support of large scale Data: Generated large scale graphlayout map

Rating: 5

12.1.7. Impact Analysis and Reporting

The *ARIS IT Architect* provides different ways for supporting impact analysis computations on the information stored within the repository. From these ways, the usage of the relationship tab in the property dialog can be considered to be simplest one to evaluate, which objects are directly impacted (while related) by a given one. The properties dialog is available from both the hierarchical project explorer as well as from a context menu on a given object representation, which is available on every graphical model. Making repeatedly use of this facility, the user can also execute transitive impact analysis. From there the impact analysis results can also be displayed graphically, as the related objects can be dragged to a diagram, where subsequently the mechanims for automatically displaying relationships can be used.

Another, way for performing impact analyses is provided by the *ARIS IT Architect* using the embedded query wizard, see Figure 12.12. Thereby, the user can poll all objects directly related to a given object. Another type of predefined queries allows to join two queries in a way, that the result delivered by the first query is used as input to the second one. Utilizing this mechanism, the user can perform arbitrary transitive impact analysis computations. As this querying mechanism needs a skilled user to take full advantage of provided expressiveness, the *ARIS IT Architect* offers the option to store a previously created query, such that any user can access it via the search dialog.

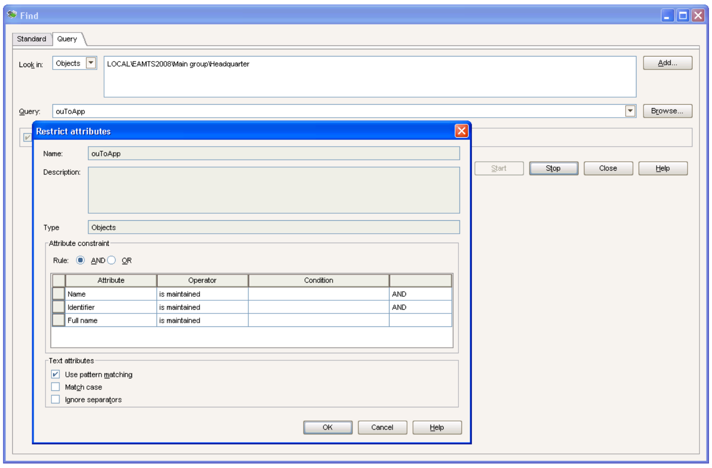


Figure 12.12.: *ARIS IT Architect* - axis Impact Analysis and Reporting: Query wizard

The scripting capabilities provide the most powerful means for define impact analysis computations, although it needs a skilled user to use them. These abilities can be leveraged from within the *ARIS IT Architect* using the integrated script assistant. Nevertheless, while this method allows full-scale access to information, the usage of this mechanism calls for a skilled user with some experience in defining script queries. This method of querying is especially neccessary, if more complex computations should be employed in querying and analyzing the data within the repository, e. g. by calculating aggregated values.

The results of impact analysis on the information from the repository can be published making use of various integrated report facilities. Thereby, the information can be exported to different formats, like Microsoft Word, *HTML*, Microsoft Excel, or *PDF* leading to a report as shown in Figure 12.13.

Rating: 5

12.1.8. Usability

In the *ARIS IT Architect* all objects are organized in a user definable hierarchic structure, represented by folders. The integrated functionalities as the *Designer*, *Matrix Editor*, and *Explorer* can be reached comfortably via shortcuts from the main window of the tool. The data governed within the repository is organized in different *databases*, which can be managed consistently from the user interface of the *ARIS IT Architect*.

Object name	Object type	Relationship type	Object name	Object type
Business Traveling System	**Application system**			
Group	Main group			
Creator	system			
Identifier	1020.0			
User attribute Text 121	5.0			
Last user	system			
User attribute Text 120	0.992			
User attribute Text 2	1.5			
Last change	Nov 12, 2007 8:43:17 AM			
User attribute Int 12	11			
Time of generation	Nov 12, 2007 8:37:44 AM			
Maintenance Costs	0.0			
		is under responsibility of	Subsidiary Munich	Organizational unit
			Group	Main group
			Last user	system
			Identifier	110002.0
			Last change	Nov 12, 2007 8:44:44 AM
			Creator	system
			Time of generation	Nov 12, 2007 8:37:48 AM
		supports	Function	Function
			Group	Main group
			Creator	system
			Last change	Dec 4, 2007 6:11:27 AM
			Last user	system
			Time of generation	Dec 4, 2007 6:11:18 AM

Figure 12.13.: *ARIS IT Architect* - axis Impact Analysis and Reporting: Impact analysis result report

Especially, the *Matrix Editor* offers an elegant way to define and maintain relationships between objects. Inside the models it is possible to create new object and to edit the objects attributes and relationships between the objects. The *ARIS IT Architect* cares for consistence between the data and its graphical representation, while it updates all changes in all other models the changed objects occurs in. The layout characteristics of a graphlayout like diagram can be adjusted by selecting different layout mechanisms.

As the *ARIS IT Architect* has a strong backing methodology, the user has to get accustomed to this methodology in order to leverage the full-scale EA management capabilities as provided by the tool. Here administration mechanisms can help, as a *filter* can be applied, rendering parts of the methodological toolset invisible to a less skilled user. Nevertheless, the strict methodology is sometimes regarded as restrictive in achieving specific visualizations, especially, if no appropriate modeltype can be found or exists. For determining the modeltype to be used for specific visualizations, the *method manual* provides comprehensive information, which is due to its overall extent not always easy to fully comprehend. Therefore, attending trainings especially concerning the *ARIS* methodology is deemed advisable in order to take full advantage of the functionalities offered by the *ARIS IT Architect*. Complementing this, a user guide is provided, which can be regarded very useful for getting a quick overview on the EA management capabilities of the *ARIS IT Architect*.

Rating: 5

12.2. Evaluation of EA Management Support

This Section describes the results of the scenario simulation for EA management support.

12.2.1. Landscape Management

The *ARIS IT Architect* provides two different ways to support modeling different versions of the application landscape. The first one is provided utilizing the concept of *building blocks*[8], which represent a tuple of business process and organizational unit. The second way relies on the *variant* mechanism, which can be used highly flexible, not only in the context of the landscape management. Subsequently, we detail both ways, starting with the building block facility.

A building block is automatically created in a process support map, representing a certain tuple of business process and organizational unit. Such a building block can be assigned a set of application systems, which provide support to the corresponding business process at the distinct organizational unit. This assignment can further be parametrized with information on its lifecycle status, i. e. a dialog allows the user to select that a specific supportive application system will be *in planning*, *running*, *in retirement*, or *retired* at a certain point in time. This information can be assigned individually to the different application system to building block relationships, thus enabling a fine grained landscape planning for current, planned, and target landscpes. In addition, this information can be visualized by colorcoding in a process support map as the one shown in Figure 12.14. In order to facilitate landscape management, the user can also select the kind process support map (*as-is*, *plan*, *to-be* or *complete*) and how it would look like at a selected point in time, which can be conveniently chosen via a dialog (see Figure 12.15).

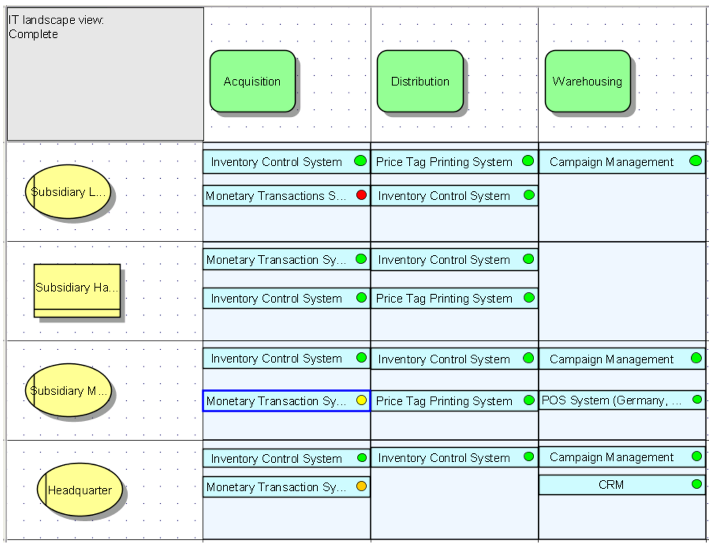


Figure 12.14.: *ARIS IT Architect* - axis Landscape Management: Process support map of the current landscape

The user can further use the building blocks in creating and maintaining relationships to projects (mapped as *Functions* - cf. Section 12.2.3), which are the drivers for changes in the application landscape. Therein, it is possible, to assign a project to a building block and to use the project's

[8]In German: *Bebauungseinheit*.

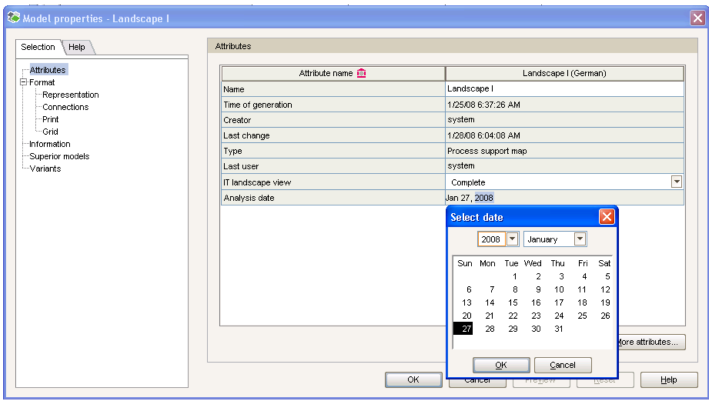


Figure 12.15.: *ARIS IT Architect* - axis Landscape Management: Dialog for selecting the report time for a process support map

relationships *phaseIn* and *phaseOut* to reference the distinct application system, which is affected by the distinct project. Nevertheless, this relationships have to be created and maintained manually, which might cause effort in keeping the information consistent[9].

As a second way to model different version of the application landscape, the *ARIS IT Architect* provides the *variant* mechanism. Every object or model can be assigned different variants representing different versions, as e. g. current, planned, or target. This mechanism makes it possible to plan and maintain distinct future states of the application landscape, with the additional potential to enrich this state information with data about the period of validity for the state. Nevertheless, this enrichment has to employ an user-defined attribute and cannot directly be derived from information about projects, which drive the transformation of the application landscape. As a consequence, the linkage between the planning states of the landscape and the projects causing these future states has to be maintained manually or implemented using the integrated scripting engine of the *ARIS IT Architect*.

The creation of visualizations depicting different states of the application landscape is possible. The user can create different variants of the model, displaying the application landscape, and use automated color coding therein to highlight applications, which are affected by change over time.

By utilizing the *ARIS* variant mechanism, it is also possible to model and maintain different potential application landscape scenarios for a given future point in time. Choosing one of these scenarios to be the actual planning state is directly possible utilizing user defined attributes for indicating validity as outlined above. A mechanism for versioning and historizing current, planned, or target landscapes modeled at a given point in time and continually updated is not directly provided by the *ARIS IT Architect*, but could be implemented using the solution approaches as provided in Section 12.1.5.

To report changes between the current, the planned, or target application landscape, which are modeled using the variants concept of the *ARIS IT Architect*, the tool provides a wizard for creating comparisons between these model variants. The user can select, which kind of changes should be reported, e. g. ongoing existence of objects or attribute changes. This comparison results in a tabular report as shown in Figure 12.16, which displays the evolution of the application landscape. Another powerful utility for comparing model variants is the graphical comparison facility, which allows to display the changes by highlighting the objects in a graphical representation of the model as shown in Figure 12.17. This graphical comparison is based on the tabular report facility, so it is easy to create both report and visualization in one step.

[9] According to IDS Scheer, the upcoming version 7.1 of the *ARIS IT Architect* will include a specific dialog facilitating such maintenance actions.

Nevertheless, as the variants are not connected to the corresponding projects, it is not directly possible to determine, which project actually caused which changes within the application landscape.

	Object name	Object type	Reference model	Comparison model	Attribute comparison	Relationship comparison
2	Warehousing	Funktion	X	X	identical	identical
3	Warehouse	Organisationseinheit	X	X	identical	identical
4	Subsidiary Munich	Organisationseinheit	X	X	identical	identical
5	Subsidiary London	Organisationseinheit	X	X	identical	identical
6	Subsidiary Hamburg	Organisationseinheit	X	X	identical	identical
7	Purchasting	Funktion	X	X	identical	identical
8	Product Shipment System	Anwendungssystemtyp	X	X	identical	identical
9	Price Tag Printing System	Anwendungssystemtyp	X (3)	X (3)	identical	identical
10	POS System (Great Britain)	Anwendungssystemtyp	X	X	identical	identical
11	POS System (Germany, Munich)	Anwendungssystemtyp	X	X	identical	identical
12	POS System (Germany, Hamburg)	Anwendungssystemtyp	X	X	identical	identical
13	Online Shop	Anwendungssystemtyp	X	X	identical	identical
14	Monetary Transactions System (Greate Britain)	Anwendungssystemtyp	X (2)			
15	Monetary Transaction System (Germany)	Anwendungssystemtyp	X (6)	X (3)	identical	identical
16	Invoice verification	Funktion	X	X	identical	identical
17	Inventory Control System	Anwendungssystemtyp	X (12)	X (12)	identical	identical
18	Headquarter	Organisationseinheit	X	X	identical	identical
19	Goods receipt	Funktion	X	X	identical	identical
20	Distribution	Funktion	X	X	identical	identical

Figure 12.16.: *ARIS IT Architect* - axis Landscape Management: Tabular report of model comparison

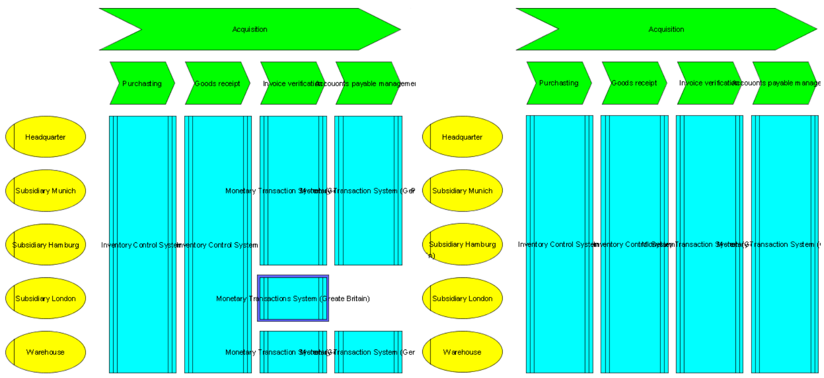


Figure 12.17.: *ARIS IT Architect* - axis Landscape Management: Graphical report of model comparison

Procedure consistency is mostly given. The concepts from the *SoCaStore* information model can be mapped appropriately and the deliverables can be achieved, although information on past planning states cannot be accessed directly.

Procedure integration is completely given. Information previously entered into the tool can be reused in this context.

Rating: 6

12.2.2. Demand Management

The *ARIS* methodology does not provide direct support for demand management, hence no objecttype for representing demands exists in the preconfigured information model. Nevertheless, demands can be mapped to the *ARIS* objecttype *Technical Term*, while projects can be modeled as *Functions*[10]. Thereby, it is possible to map the relationships as demanded in this scenario, to existing relationships of these object types to the type *Application System*.

Having performed the mapping, as alluded to above, received demands can be created using the *ARIS IT Architect* object explorer. The user can further leverage functionalities as grouping objects into a folder hierarchy, for introducing structure to the demands governed in the tool.

The deduction of projects based on the received demands can be performed graphically employing a model. The demands can be dragged from the explorer to the model and new projects can be created graphically. Also the relationships that indicate, which projects result from which demands, can be created graphically in the model by dragging a connection between the involved demands and projects. An examplary model is shown in Figure 12.18, depicting two projects deducted from the incoming demands and the affected application systems.

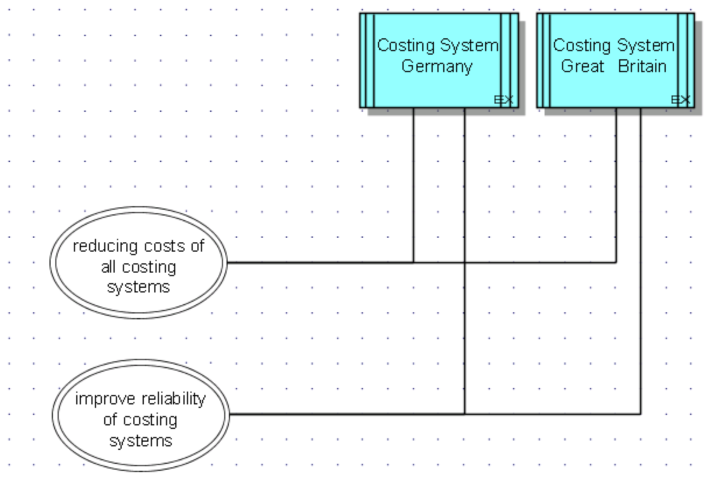


Figure 12.18.: *ARIS IT Architect* - axis Demand Management: Example model of demands, resulting projects, and affected application systems

The *ARIS IT Architect* offers multiple possibilities to query, which application systems are affected by which demands. One, not fully automated way is to use the properties dialog of the relevant demand objects. The dialog enables the user to trace the relationships between the objects to reach the application systems affected by a selected demand. A more automated way can be achieved by creating a separate model and dragging all demands, which should be examinated at the model, to the model's canvas. By utilizing the relationship dialog from a modeled demand, all referenced functions and from there all referenced application systems can be automatically shown in the diagram. At last, the most automated way is to create a user defined query, which opperates on *Demands* and trace the relationship over the projects to the affected application systems automatically. This user defined query can be called from any context, where a demand is modeled or especially from the *ARIS*

[10] According to IDS Scheer, *demands* can also be mapped to *Objectives*, utilizing a user defined attribute for distinguishing them from business level goals, which are also mapped to *Objectives* (see Section 12.2.5. Further, according to IDS Scheer, *ARIS IT Architect* 7.1 will provide distinct objecttypes and modeltypes for managing demands.

explorer. The execution of the query results in a list of all affected application systems, of which an example is shown in Figure 12.19.

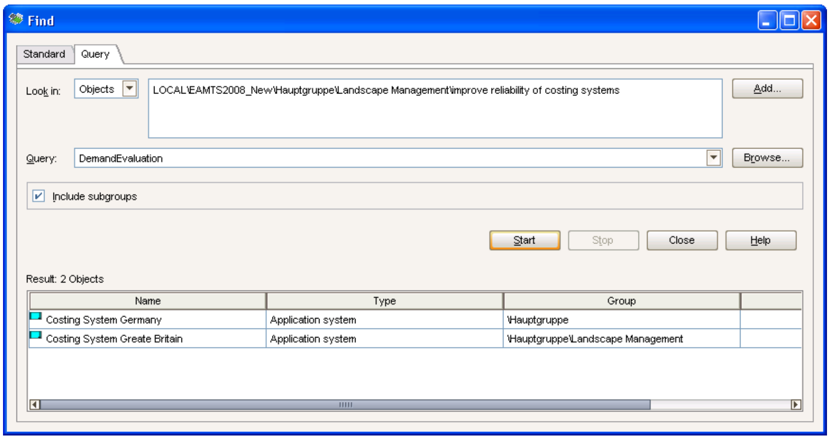


Figure 12.19.: *ARIS IT Architect* - axis Demand Management: Example query of application systems affected by demands

Procedure consistency is mostly given. Demands have to be mapped to concepts of the predefined infomation model, which may not be appropriate under all circumstances.

Procedure integration is completely given. Information already present in the tool, especially concerning the application systems, can be reused in this context.

Rating: 6

12.2.3. Project Portfolio Management

As described in Section 12.2.2, an objecttype representing a project is not part of the *ARIS* methodology as contained in the *ARIS IT Architect*[11]. As a consequence, projects are mapped to *Functions*. As the objecttype *Function* supports most of the attributes required in this scenario, the mapping can be considered to be appropriate.

In executing project portfolio management, many of the concepts from the demand management can be reused. As an example, the project proposals can be structured into a folder hierarchy, to leverage the quick overview. Also the impact analysis that shows, which organizational units are affected by a project proposal, can be performed in a way similar to the one described in demand management. Therefore, the user can make use of the properties dialog to transitively traverse the relationships from functions to application systems and organizational units. Again, the most automated way is to create a user defined query, which starts at a selected project proposal, traverses the relationships over the application systems and delivers a list of affected organizational units. An examplary outcome of such a query is shown in Figure 12.20.

Reports that evaluate the projects' urgency, costs, and benefits leading to diagrams, which show the results of these evaluations for example as bar charts are not directly supported in the *ARIS IT Architect*. Nevertheless, they can be created by defining them as new reports using the integrated

[11]According to IDS Scheer, the *Business Optimizer* can be considered as the tool providing strong support for project portfolio management. Nevertheless, this tool was not evaluated in the survey.

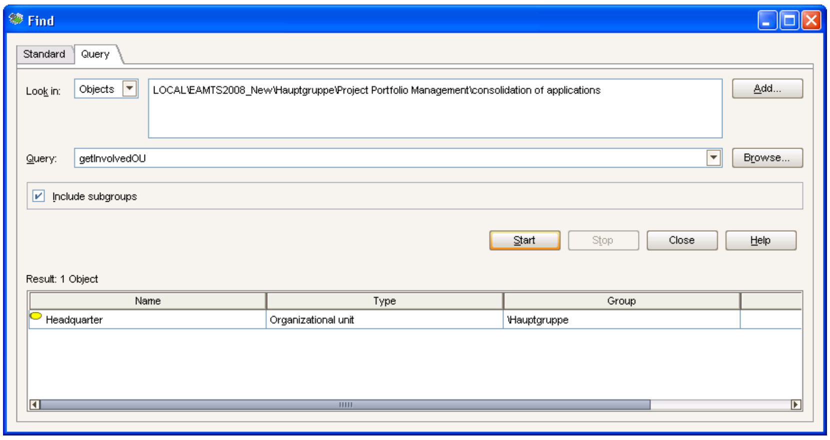


Figure 12.20.: *ARIS IT Architect* - axis Project Portfolio Management: Example query of organizational units affected by project proposals

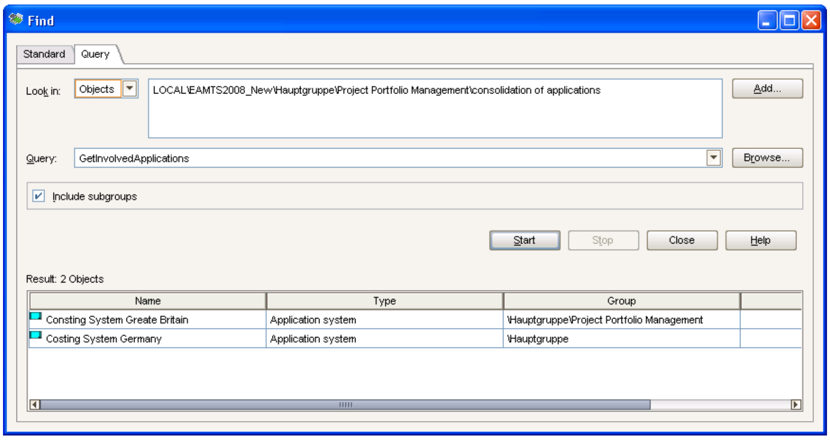


Figure 12.21.: *ARIS IT Architect* - axis Project Portfolio Management: Example query of application systems affected by project proposals

scripting engine. Another possibility for creating similar reports is provided via the tight integration to Microsoft Excel, which the necessary data can be exported to. Finally, a project portfolio matrix can be generated using a predefined report and adapt the corresponding configuration file, as explained in Section 12.1.2.

Since the *ARIS IT Architect* does not offer aggregate functions in querying, it is not directly possible to automatically determine and report on the number of project proposals affecting an application system. Nevertheless, by using the scripting functionalities as offered by the tool, a skilled user can easily implement such a functionality.

Procedure consistency is mostly given. Most concepts can be mapped to the information model appropriately, the deliverables can be created, although some customization is necessary.

Procedure integration is mostly given. Information previously entered into the tool can be reused, although no direct transformation from demands to projects is supported.

Rating: 5

12.2.4. Synchronization Management

According to IDS Scheer, the *ARIS IT Architect* should not be considered a full-scale project management tool. Therefore, particularly aspects of project synchronization do not fall into the focus of the tool. No objecttype for representing a project is provided in the *ARIS* methodology, therefore a mapping to *functions* was considered to be appropriate during the evaluations of Project Portfolio Management. The same is true in the context of synchronization management, although no direct support for the concept of project delays can be realized utilizing the predefined information model, such that considerations on the impact of a delayed project would employ usage of user defined attributes[12].

By querying the dependencies between the objects, it is possible to perform an impact analysis for determining project interdependencies. The user can therefore utilize the relationship dialog of a function to traverse to the affected application systems first. From there, the relationship dialog of the application system can be used to list all projects, which affect the selected application system. This possibility to perform an impact analysis is complemented by the graphical modeling facilities of the *ARIS IT Architect*. Using them, the user can create a diagram displaying a project an automatically showing all related objects, although some manual interaction is needed, since only directly related objects can be automatically added to an existing visualization.

The most automated way is the creation of an user defined query, which starts from a designated project, traverses the relationships to the involved organizational units or to the affected application systems, and finally traverses the relationships back to all projects, that involve the corresponding organizational units, or affect the applications systems. An example for a query dialog is shown in Figure 12.22.

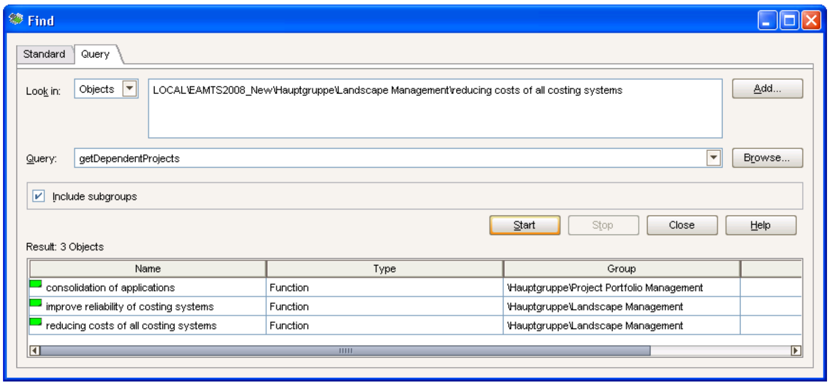


Figure 12.22.: *ARIS IT Architect* - axis Synchronization Management: Example query of depending projects

A diagram that illustrates in which time intervalls application systems are affected by projects, can be created employing some manual interaction, leading to a visualization as shown in Figure 12.23. A similar visualization could be created automatically, if the *Lifecycle report* as shown in Figure 12.6 was adapted appropriately.

Procedure consistency is mostly given. Most concepts can be mapped to the predefined information model except for concepts exerting temporal dependencies.

Procedure integration is mostly given. Information, which has been entered in the tool, can be reused in this scenario except for temporal information.

Rating: 4

[12]According to IDS Scheer, the upcoming version 7.1 of the *ARIS IT Architect* will include further functions concerning modeling of temporal dependencies.

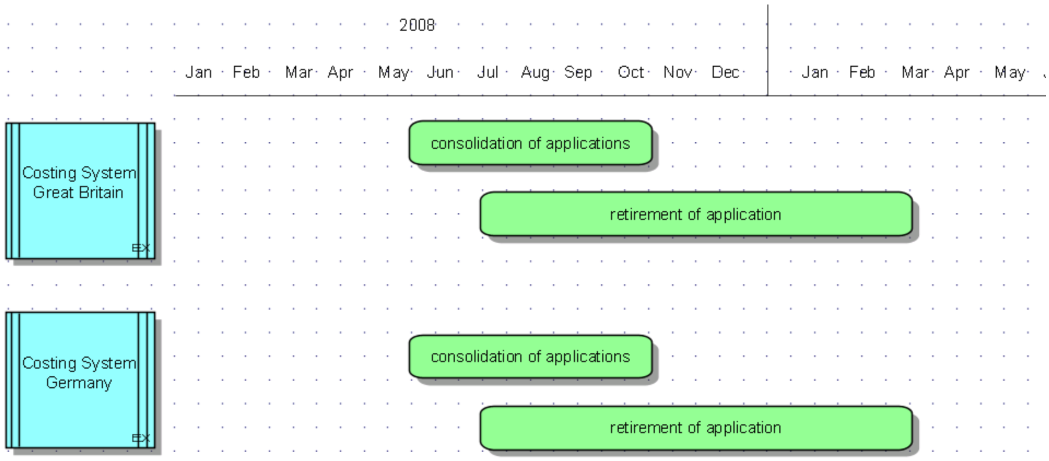


Figure 12.23.: *ARIS IT Architect* - axis Synchronization Management: Example of a time interval map that shows in which time period an application system is affected by which projects

12.2.5. Strategies and Goals Management

The *ARIS* information model defines distinct objecttypes to represent strategies and objectives, of which the latter are used to represent the goals as employed in this scenario. Instances of these objecttypes can be modeled on a special modeltype, called *cause and effect diagram*. An example of such a diagram is shown in Figure 12.24, depicting the strategy *gaining market leadership* together with its related objectives.

The *ARIS* methodology provides the concept of *assignment*, which can be used to decompose an objective into more detailed ones. Therein, assignments are represented by independent models that are linked to the object, which is actually decomposed. An exemplary model showing an assignment decomposition of the objective *be a quality supplier* is shown in Figure 12.25.

The *ARIS IT Architect* provides several possibilities to determine, which strategy leads to which goals. One of them is to generate a dedicated *cause and effect diagram* for the strategy, similar to the one already shown in Figure 12.24. The wizard guiding through the generation allows to select all related objects of the given strategy to be displayed in the generated diagram. Another possiblity would be to make use of the relationship dialog, which all related objectives of the selected strategy are displayed in.

In order to evaluate, which goals corresponding to a given strategy have been reached and which organizational units have reached their goals, the information model of the *ARIS IT Architect* provides the attribute *actual value* for the objecttype *objective*. A user defined query can perform computations on this attribute and can thus select the goals, of which the *actual value* does not match a predefined value. Based on the result of this query, a second query can be used to traverse the relationship from the selected goals to the affected application systems and the involved organizational units. Composing this two queries, the user can quickly get a list of organizational units connected to a given strategy, which have not yet reached their goals.

For distinguishing between fulfilled and yet not fulfilled goals in a *cause and effect diagram*, the user can perform a *Balanced Scorecard Planned Actual comparison*, which uses the information on the actual and the planned value regarding an objective, to automatically apply a color-coding. Beyond this, a textual annotation, which displays the current value of the attribute *actual value*, could also be created for every objective.

Figure 12.24.: *ARIS IT Architect* - axis Strategies and Goals Management: Example of a *cause and effect diagram*

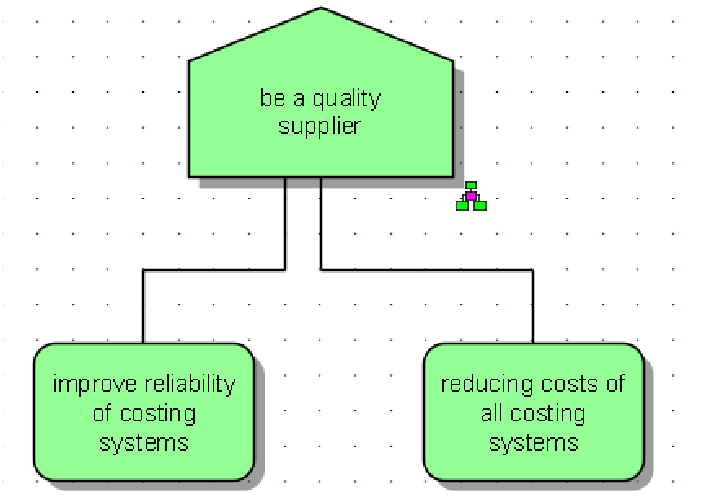


Figure 12.25.: *ARIS IT Architect* - axis Strategies and Goals Management: Example of a assigment model which details the objective *be a quality supplier*

Procedure consistency is completely given. The concepts of strategies and goals are appropriately represented in the information model of the tool.

Procedure integration is completely given. Information previously entered into the tool can be reused in this context.

Rating: 6

12.2.6. Business Object Management

The exchange of business objects in the application landscape can be modeled in the *ARIS IT Architect* utilizing the *access diagram*. This modeltype allows to model application systems as well as the corresponding business objects, which are mapped to *information flow* objects in the methodology. Another mapping would also be possible, not explicating business objects in the diagram, but modeling the exchange of information between two application system via the *trasmitsDataTo* relationship. For this mapping the modeltype *context diagram* can be used.

If any of these mappings is applied, information concerning the business objects can be handled. Subsequently, the first mapping is further detailed. The application systems provides the relationships *send to* and *receive from*, which can be used to model information flows. Figure 12.26 shows an exemplary visualization, displaying the producer and consumer relationship regarding a business object.

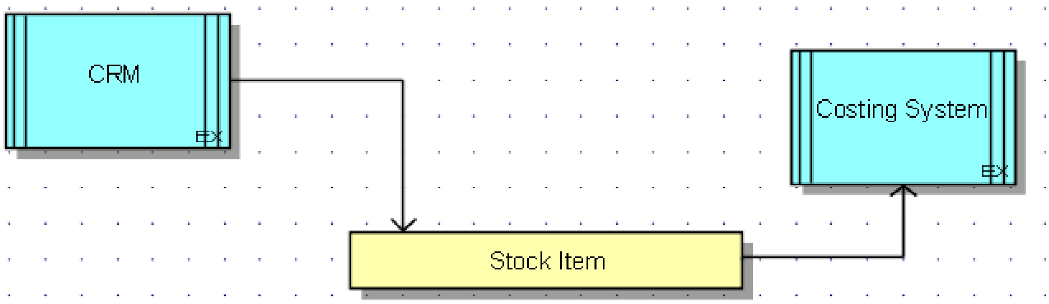


Figure 12.26.: *ARIS IT Architect* - axis Business Object Management: Example of an information flow model

The *information flow* objects can be queried utilizing a user defined query, which traverses the *received from* relationship, in order to create a list of application systems that modify a selected business object. Furthermore, a user defined query can be used to traverse the *send to* relationship, listing the application systems consuming a certain business object. Figures 12.27 and 12.28 show exemplary results of queries of that kind.

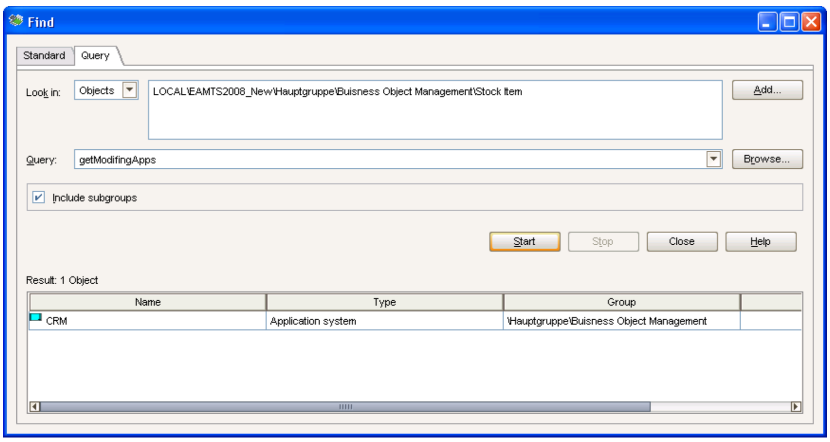


Figure 12.27.: *ARIS IT Architect* - axis Business Object Management: Example query of modifying application systems

Further, a user defined attribute can be used to model, which business object needs manual processing in business process execution. Subsequently, a user defined query can be executed, comparing the

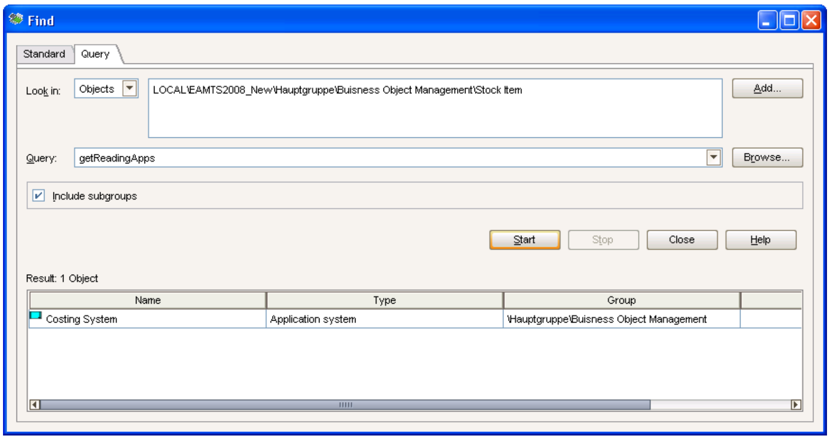


Figure 12.28.: *ARIS IT Architect* - axis Business Object Management: Example query of reading application systems

value of the user defined attribute to a selected value, resulting in a list of all objects having the selected property. This especially could be used for determining the business objects, which need manual processing.

The alternative way to manage business objects in the *ARIS IT Architect* is, as alluded to above, using the *context diagram* for creating the *transmitDataTo* relationship between two application systems. Having created such a relationship, the user can model the details of the data transmission represented thereby using the *program flow chart*. In this modeltype, an instance of objecttype *Cluster/Data Model* are used to represent a business object, which is exchanged between two application systems referenced by the *inputFrom* and *outputTo* relationships. Furthermore, information on the *Protocol* used for exchanging the business object can be added to the diagram. Finally, a predefined report *Display Data Flows* can be used to aggregate the information on the business objects exchanged between a set of application systems, if the information is modeled in the way as described above.

Procedure consistency is mostly given - an appropriate mappping for the concept *Business Object* can be found in the information model.

Procedure integration is completely given. Information previously entered into the tool can be reused in this context.

Rating: 5

12.2.7. SOA Transformation

For supporting the selection of application systems, which are due to their number of users appropriate candidates for transformation into services, the *ARIS IT Architect* provides the possibility to use the *number of internal users* attribute, which is available for the objecttype application system. If this attribute is maintained with the number of daily users, it can be used to create a process support map holding the textual information of daily users per application system. An example of such a process support map is shown in Figure 12.29.

An automated annotation of visualizations depicting application systems, depending on the *number of internal users* attribute is not directly supported by the *ARIS IT Architect*, but could be leveraged utilizing the scripting engine. Beyond this, it is possible to apply this color-coding manually, leading to a visualization as the one shown in Figure 12.30. This colorcoding could be automatically created via adapting the standard lifecycle report (see Section 12.1.3).

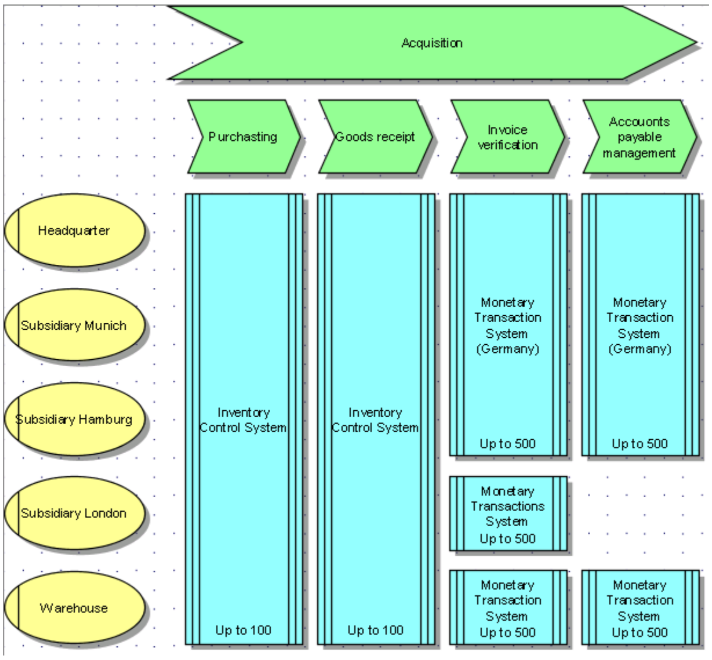


Figure 12.29.: *ARIS IT Architect* - axis SOA Transformation: Example of an process support map that shows the number of daily users of the application systems

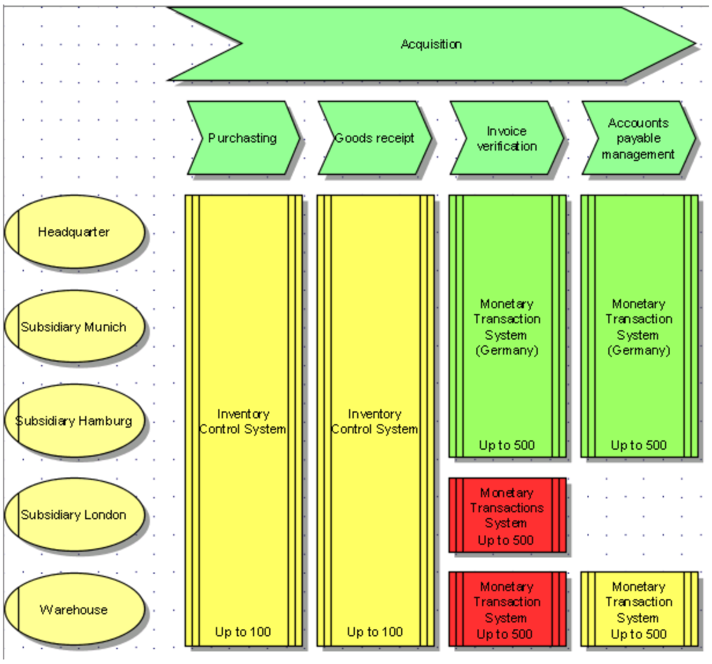


Figure 12.30.: *ARIS IT Architect* - axis SOA Transformation: Example of a process support map that indicates the number of daily users by color-coding

A time interval map displaying temporal information on the application systems and the migration to services can be created via adaptation of the *Lifecycle report* (cf. Figure 12.6) , although it is not directly possible to visualize, that functionalities are moved from an application system to a newly introduced service at a certain point in time.

A tabular report, showing the application systems, corresponding business objects and their interaction can be created relying on the export to Microsoft Excel. Thereby, the user can chose to export the application systems and their directly related objects. Furthermore, the reporting facilities of the *ARIS IT Architect* could be used in creating such a report. If further information on the impact of the SOA transformation on the current application landscape is needed, the full power of the model comparison facilities, as described in Section 12.2.1, can be leveraged.

Procedure consistency is mostly given. User defined attributes can extend the information model at points where a direct mapping cannot be found.

Procedure integration is completely given. Information, especially from landscape management, can be reused in the context of this scenario.

Rating: 5

12.2.8. IT Architecture Management

The management of architectural standards is supported by the *ARIS IT Architect* relying on a concept called *quadrants*. These quadrants can be used for performing a classification on the application systems according to their underlying architectural solution or blueprint, although the latter concepts are not directly present in the information model of the *ARIS IT Architect*. Nevertheless, they can be mapped to *application system classes* and therefore be represented in the tool.

Having mapped solutions (and/or blueprints) to application system classes, the user can assign actual application systems to these classes, utilizing the quadrants mechanism, as alluded to above. Therein, the user employs the capability of the *ARIS IT Architect* to create or maintain relationship between objects, which are nested in certain modeltypes. Such an assignment leads to a visualization as the one shown in Figure 12.31. This visualization helps to get an overview, which application system is based on which architectural solution.

If the information about the underlying architectural solutions of the application systems is maintained employing the mechanisms explained above, the user can display this information on an automatically created process support map. This map shows the assignment of an application system to an organizational unit and a business process by utilizing relative positioning and can also display information on the architectural solution of the application system by color coding the respective symbol. An exemplary visualization of that type is shown in Figure 12.32.

Complementing the diagrams, as alluded to above, is the definition of a user defined query, which returns a list of all architectural solutions used at a given organizational unit. This query starts at the objecttype organizational unit and traverses the relationship to the application systems and from their to the respective architectural solutions. Furthermore, a user defined query can be created, displaying, which application systems are based on which architectural solutions, by starting at the *application system class* and traverses to the associated application system. Figure 12.33 shows an example of a query like this.

Finally, an overview on the used architectural solutions can be created employing a search over the repository data, restricted to the objecttype *application system class*. The same considerations as provided above can also be applied to architectural blueprint, which therein had to be mapped to objects of type *application system class*. With these two mappings similarly performed, it is further possible to rely on the functionality of the so-called *architecture construction kit*. Thereby, the user can detail tiered architectures in the *IT architecture matrix* model (see Figure 12.34), which is due to the corresponding mapping of architectural blueprints and solutions applicable to both abstract and realization level concepts. Such a matrix can subsequently be associated to an application system type further facilitating reflections on the constituents of an architectural element.

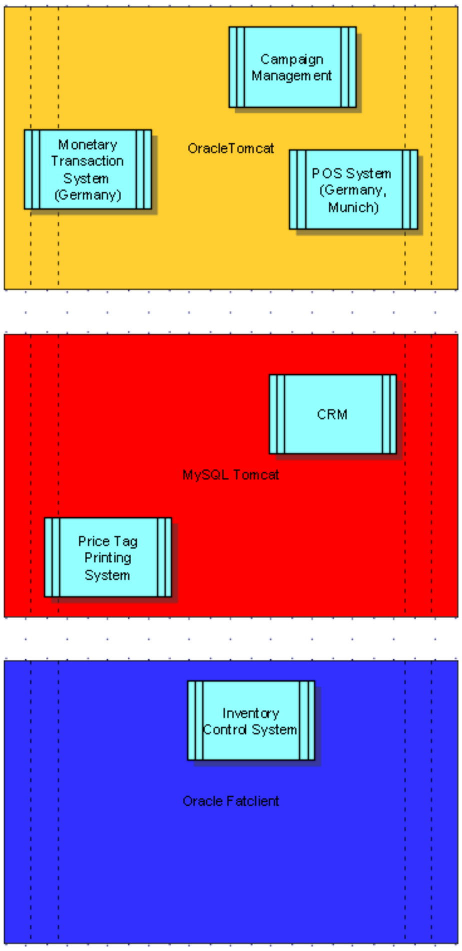


Figure 12.31.: *ARIS IT Architect* - axis IT Architecture Management: Example of an application system diagram depicting, which application system is assigned to which architectural solution

Procedure consistency is mostly given. Strong capabilities for automatical creation of visualizations are available, although the mapping of architectural blueprints and solutions to the same information model concept may cause minor inconsistencies.

Procedure integration is completely given. Information, especially from landscape management, can be used in the context of this scenario.

Rating: 6

12.2.9. Infrastructure Management

The *ARIS IT Architect* does not directly support a multitude of different infrastructure component concepts, but the concept of the *database management system*[13] is present in the information model, supplying all neccessary relationships to other information model concepts. Based on this objecttype

[13] According to IDS Scheer, this objecttype should no longer be used – application system types are to be used for modeling infrastructure components. From this mapping a consistency concern similar to the one alluded to in Section 12.2.8 might arise.

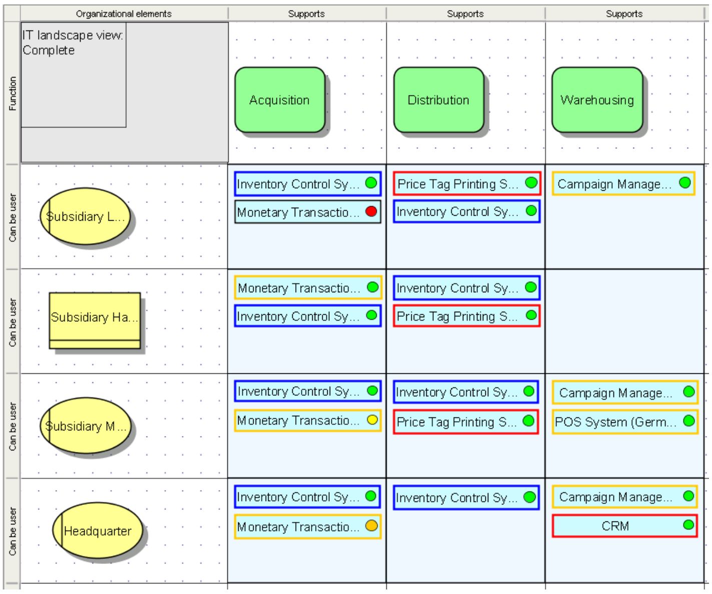


Figure 12.32.: *ARIS IT Architect* - axis IT Architecture Management: Example of a process support map visualizing the application systems indicating their architectural solutions by colorcoding

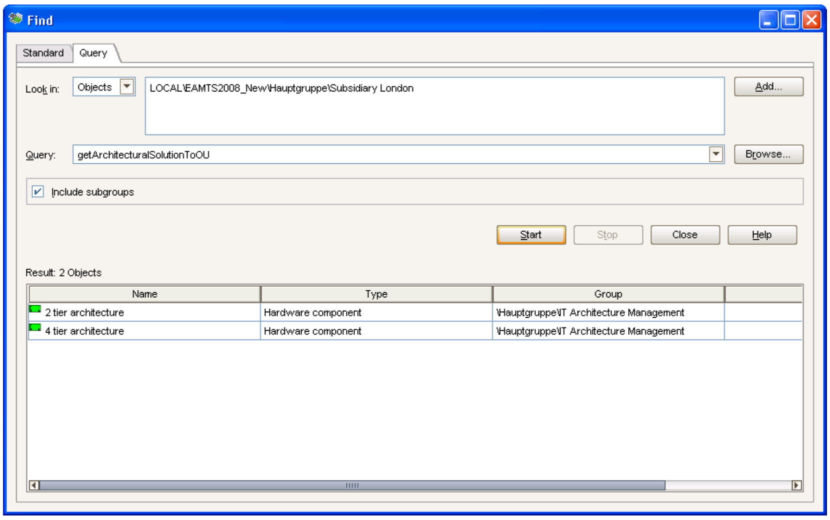


Figure 12.33.: *ARIS IT Architect* - axis IT Architecture Management: Example of a query that evaluates, which architectural solution are used at the given organizational unit

or respectively using application system type, it would be possible to introduce further infrastructure components, relying on a user defined attribute for distinguishing these types.

In order to get an overview on the database managent systems running out of support, the *ARIS IT Architect* can supply a tabular report, a time interval map can be created utilizing the *Lifecycle report*. Further, the *ARIS* methodology contains a modeltype, which can be used to automate the visualization of start or end of support dates, leading to a diagram as shown in Figure 12.35. The latter

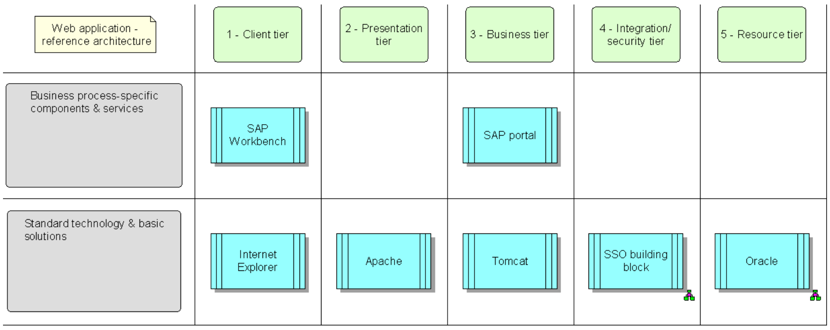


Figure 12.34.: *ARIS IT Architect* - axis IT Architecture Management: Example of an architecture matrix describing tiers and levels of a reference architecture

functionality can additionally be used to display operation or licensing costs in a textual annotation, while a color-coding of the corresponding symbols would employ the creation of an appropriate user defined script.

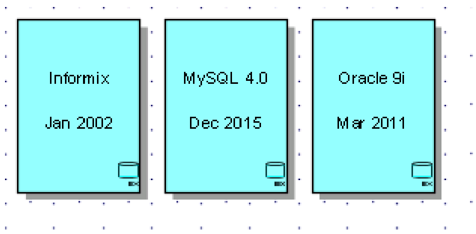


Figure 12.35.: *ARIS IT Architect* - axis Infrastructure Management: Example of a diagram with textual annotation of the end of support period

When concerning the relationships between the application systems and the corresponding database management systems, the user can utilize a predefined reference in the information model. This reference can subsequently be used in visualizing the relationship in an automatically layouted diagram as shown in Figure 12.36. Another option for getting an overview on the application system to database management system relationship is the creation of a user defined query, traversing the corresponding relationship *based on* connecting the database management system and the related application system. The execution of the query leads to a visualization, of which an example is shown in Figure 12.37. This query can further be extended to also include the organizational units (linked via the *host at* relationship), such that the user can determine, which organizational units are affected, if a distinct database management system is to be retired.

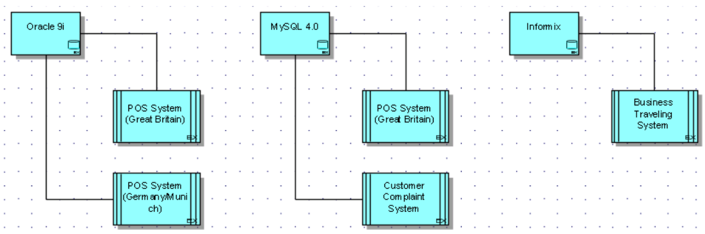


Figure 12.36.: *ARIS IT Architect* - axis Infrastructure Management: Example of a diagram showing the application systems related to a database management system

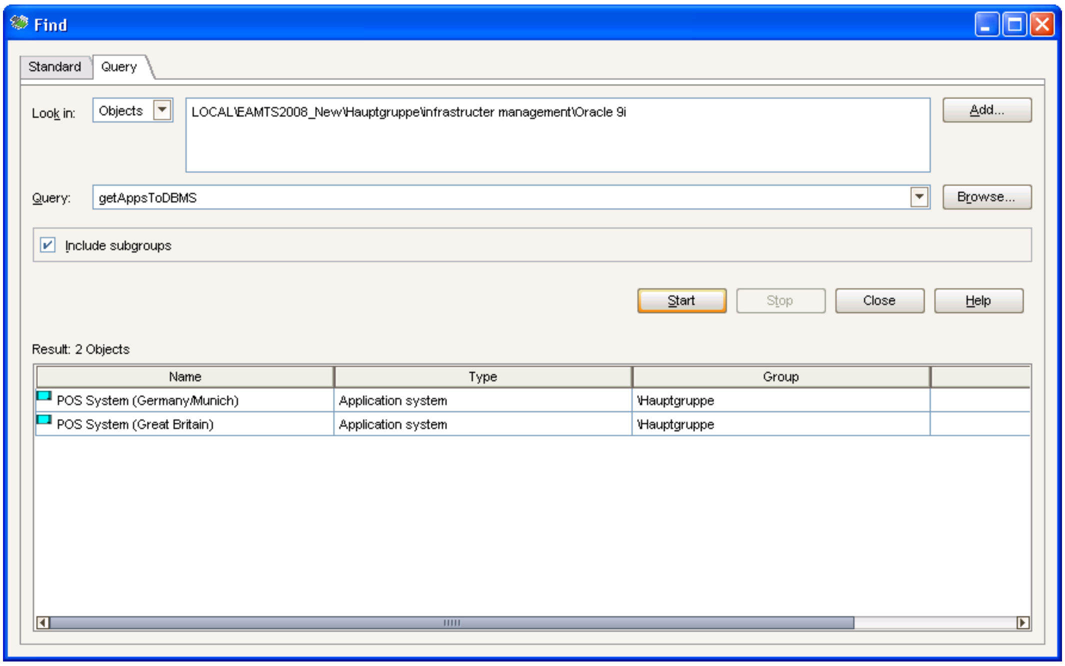


Figure 12.37.: *ARIS IT Architect* - axis Infrastructure Management: Example of a query evaluating, which applications systems depend on a given database management system

If finally an overview on the database management systems currently operated throughout the enterprise should be created, the user can rely on the search functionality of the *ARIS IT Architect*, further restricting the result set to objects of type *database system*. The same would be true for infrastructure components of other types, while here the user defined discriminator attribute, as alluded to above, would have to be taken into consideration for filtering.

Procedure consistency is mostly given. Database management system can be mapped to a corresponding concept in the information model, which also provides a good starting point for mapping other infrastructure components[14].

Procedure integration is completely given. Information, especially from landscape management, can be reused in the context of this scenario.

Rating: 5

[14] If, according to IDS Scheer, application system types were used for the mapping, the user would have to manually take care of the distinction between *business* and *technical* application system types.

12.3. Tool Vendor's Profile

Dirk Maurer, Product Manager ARIS Design Platform

IDS Scheer AG

The software and consulting company IDS Scheer is the worldwide leading provider of Business Process Management and IT solutions. With the ARIS Platform, IDS Scheer offers a complete tool portfolio for "Business Process Excellence", which is highly integrated both technologically and from the content aspect. This encompasses methods, software and solutions for all phases of the process lifecycle - from strategy and design through to implementation and controlling. ARIS Platform is thus a clear unique selling point for IDS Scheer and provides customers in corporations and public authorities with software support for their whole process lifecycles and management of IT architectures. Within the ARIS Platform, ARIS Toolset and ARIS Business Architect are the world's most frequently purchased tools for process optimization. Under a strategic cooperation with SAP, the ARIS tools and methods will in future be standard in the SAP NetWeaver platform. ARIS SmartPath is a solution that will make rapid SAP introduction a reality for medium-sized companies as well. Thanks to the integrated approach of ARIS Value Engineering (AVE), IDS Scheer consultants view their customers' organizations holistically. AVE means building bridges between corporate strategy, the processes derived from it, the IT solutions needed to support it and also the controlling of running processes. Moreover, customers profit from complete global services for outsourcing and support. IDS Scheer was founded in 1984 by Prof. August-Wilhelm Scheer, who today is Chairman of the Supervisory Board and Chief Technology Advisor. The company now serves about 6,000 customers in over 70 countries through a network of its own branch offices and partners. In 2006, the company earned revenues of about 354 million Euro. Over 2800 people are working for IDS Scheer worldwide. The company is listed in the TecDAX on the Frankfurt Stock Exchange (Germany).

ARIS Solutions

ARIS Solutions combine selected ARIS products with ARIS Value Engineering procedure models designed to meet specific needs. These specialist solution portfolios have proven successful in numerous customer projects. Leveraging this experience enables fast, cost-effective creation of customer solutions. All solutions are based on the central, integrated ARIS repository, so data can be exchanged and processed across multiple solutions. The entire targeted user groups - organizational department and management, operational and IT departments, and financial controllers - work in an integrated environment. Integrated process management is therefore possible throughout the enterprise.

ARIS Solutions areas are:

- Enterprise BPM
- Enterprise Architecture Management
- Process-driven SAP Implementation
- Service Oriented Architecture
- Process Intelligence and Performance Management
- Governance, Risk and Compliance Management

ARIS Solution for Enterprise Architecture

Corporate IT resources constitute a complex system. An enterprise architecture (EA) describes this system and establishes standards for creating and updating it. Understanding the system requires a number of views-business processes, organizational structure, applications, data, interfaces, and technologies. Multiple perspectives must be taken into account, ranging from the enterprise view at a highly abstract level to detailed views of individual business units, design aspects, and physical systems and devices. With the ARIS Value Engineering procedural model and ARIS IT Architect, this solution combines market-leading BPM technologies with global EA standards. It enables customers to plan, visualize and evaluate technologies, processes, data and organizations through a central repository, eliminating redundancies. The consistent integration of all data in the ARIS Solution for Enterprise Architecture Management means that architecture elements can be reused across all views in IT Landscape Planning, IT Architecture Management, and IT Governance scenarios.

Figure 12.38.: Logo of IDS Scheer AG

12.4. Statement concerning ARIS ArchiMate Modeler

According to the voting of the industry sponsors and partners of the survey, both ARIS IT Architect and ARIS ArchiMate Modeler should both be evaluated within the survey. Nevertheless, in accordance with IDS Scheer we decided to not perform an in-depth evaluation of both tools, but to focus on the IT Architect. Subsequently, we provide a short statement of IDS Scheer concerning the distinction and relationship between the ARIS IT Architect and the ARIS ArchiMate Modeler, followed by a short description of the ArchiMate Modeler.

> Enterprise Architecture is a topic which goes across different products of the ARIS Platform, as it comprises processes, IT, data and organizational aspects. Therefore there is not the one and only tool in the ARIS Platform which is dedicated to EA. It is a combination of tools used by different target groups. ARIS IT Architect covers the broadest range of these aspects and provides specific extensions for various IT Governance tasks, but includes the methodology and functionality of ARIS Business Architect as well. As EA is in most cases driven from IT organizations to manage the alignment of IT to the Business requirements, ARIS IT Architect is in general the best fit for this target group.
>
> ARIS ArchiMate Modeler is an extension to ARIS IT Architect or ARIS Business Architect. For organizations who want to set up an EA Management which is based on the ArchiMate framework, ARIS ArchiMate Modeler offers specific method adaptations, look & feel and a documentation how ArchiMate maps to the ARIS framework.

Manage your Business Architecture with ArchiMate

In the midst of constant technological innovation and an increasingly dynamic world market, change has become a common occurrence within organizations. In order to evolve quickly and effectively as a seamless organization, business and IT must work together.Most importantly, business and IT departments need to speak the same language. ArchiMate is a framework and model language for high-level business architecture, bridging the gap between business and IT. In conjunction with ARIS Platform, the world's most comprehensive, flexible environment in architectural management, ArchiMate forms the basis for effective integral management of business and IT.

Transformation and Architecture

Over the last few years, there has been major progress in integration technologies. The emergence of process orientation and Service-Oriented Architecture (SOA) has increased the manageability and strategic positioning of IT. At the same time, organizations are feeling increased pressure to be able to react quickly and flexibly to changing circumstances. Changes affect the entire organization - from the structure to the infrastructure. The impact of change on procedures can no longer be considered separate from the impact on IT. Therefore, it is necessary to take a holistic approach to the establishment of an organization. Additionally, organizations must clearly establish and manage the business architecture across various business areas. Different modeling methods and frameworks apply to the description and analysis of parts of the business architecture, in both IT and business management (e. g., organizational structure, operating procedures, information systems and infrastructure). ArchiMate is a consistent framework of principles, methods and models contributing to the integral recording of high-level business architecture. The purpose of ArchiMate is to explicitly describe the connection between business, applications and infrastructure. The ArchiMate standard is controlled and developed by the ArchiMate Forum, of which IDS Scheer is a member.

ARIS ArchiMate Modeler

Through ARIS ArchiMate Modeler, IDS Scheer offers the ArchiMate framework within ARIS Platform. The combination of ARIS Platform and ArchiMate provides the most comprehensive integral environment for modeling and analysis of business architecture. Simple and commonly used, ArchiMate educates both IT architects and business architects. ARIS Platform supports the complete change cycle, from strategic analysis via design and simulation to implementation, monitoring and control. Seamlessly linked, the frameworks of ArchiMate and ARIS are both developed based on the goal of superior architectural control. To implement business architecture, it is necessary to define a link from high-level architecture to detailed implementation designs. The complete palette of standard modeling techniques in the ARIS Framework provides the necessary specialization to the high-level scope of ArchiMate. The integration of ARIS Platform with execution platforms such as SAP, Oracle, IBM and Microsoft and with system management environments and CMDBs turns business architecture into an active exchange between business and IT implementation.

Key Benefits of ARIS ArchiMate Modeler

- Availability of complete ArchiMate framework
- User friendly, Web-based designer
- Architecture, as an effective guide, links strategy, design and implementation
- Single, repository for essential companywide architectural control
- Multi-user and completely scaleable
- Interactive Web publication, ability to integrate with company intranet
- Analysis via queries, reports and exports in various formats, such as XML and Excel

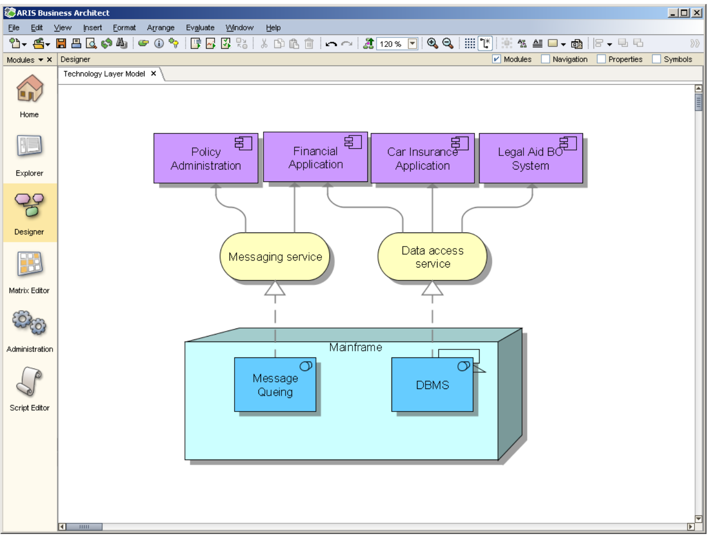


Figure 12.39.: Screenshot of ARIS ArchiMate Modeler

CHAPTER 13

MEGA International SA (MEGA Modeling Suite 2007)

Contents

MEGA Modeling Suite is a tool providing a comprehensive approach to EA management from the IT and the Business point of view, while the latter one has been a focal point in the past. Currently, a further extension of focus is under way, broadening the overall offering of MEGA SA. The tool evaluated below is *MEGA Modeling Suite* 2007.

13.1. Evaluation of Specific Functionality

Below the results of simulating the scenarios from Section 4.1 with *MEGA Modeling Suite* are detailed. Before we describe the evaluation results more in detail, the overall terminology used throughout the tool is explained and put in the context of the terminology used in the rest of the survey. Concerning the information model used, *MEGA Modeling Suite* offers different concepts of which the three most important are outlined below:

- the *MetaClass*, which corresponds to a *Class* in our information model,
- the *MetaAttribute*, which corresponds to an *Attribute* in our information model, and
- the *MetaAssociation*, which corresponds to an *Association* in our information model.

Complementing the information model concepts, *MEGA Modeling Suite* also supports the concept of the *viewpoint* on the information, as e. g. backing the visualizations used throughout the scenarios from Section 4.2. The viewpoints of *MEGA Modeling Suite* are called *Overviews* or *Diagrams*, depending on their actual intended usage.

13.1.1. Importing, Editing, and Validating

MEGA Modeling Suite supports import of data from various formats, amongst other *XMI* for instance data or Microsoft Visio for diagrams. Importing data from Microsoft Excel files is possible utilizing two different mechanisms. The first mechanism is to use the *MEGA Excel Editor*, which launches an embedded instance of Microsoft Excel comprising additional functions for synchronizing data with the MEGA repository. In this embedded instance, the user can choose a class from the information model, for which instance data should be imported. Subsequently, the editor creates a template Microsoft Excel worksheet, containing a user-defined subset of the attributes of the class (cf. Figure 13.1) to be imported. In this mechanism, transformation of data as e. g. splitting comma separated values can be achieved using the functionalities provided by Microsoft Excel. Consistency checks are partially applied, mandatory attributes, which have not been assigned a value in the Microsoft Excel worksheet, are assigned a default value.

The second mechanism for importing is to use the scripting functionalities of *MEGA Modeling Suite* as the tool offers an API to its repository, usable from VB Script (*Visual Basic*). A skilled user can then leverage the expressiveness of VBS in the import scripts in order to perform full-scale transformation operations. Nevertheless, basic importing can be performed without have to use scripting. Additionally, VBS programming is extensively used in defining constraints on the repository data, which can subsequently be used for validating the imported data. Thereby, a textual report informing about erroneous or inconsistent data can be created according to user needs. Nevertheless, the expressiveness of the scripting functions in *MEGA Modeling Suite* calls for guidance by a *MEGA Modeling Suite* consultant or a skilled programmer.

The user can easily access data within the repository using the tree-like instance navigator (cf. Figure 13.2), in which also associations can be traversed transitively. Attribute values for actual instances can be edited using a user-definable dialog. In order to leverage multiple simultaneous changes, the user has three options. The first of them leverages the functionalities of the result dialog for a search in the repository. In that dialog multiple objects can be displayed together further allowing to change their

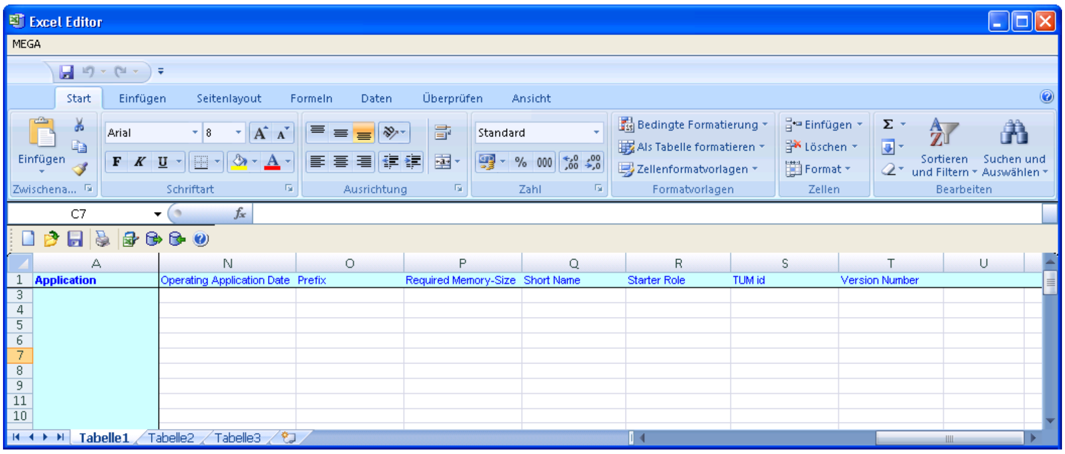


Figure 13.1.: *MEGA Modeling Suite* - axis Importing, Editing, and Validating: Template Microsoft Excel worksheet for importing applications created by the MEGA Excel Editor

attribute values. The second option to perform simultaneous changes employs the built-in scripting functionalities, while the third option is to export the data to Microsoft Excel, perform the changes there, and reimport the data into *MEGA Modeling Suite*. Data consistency is therein ensured via unique object identifiers, while nevertheless mechanisms for resolving conflicting changes can be leveraged relying on locking of objects in the repository.

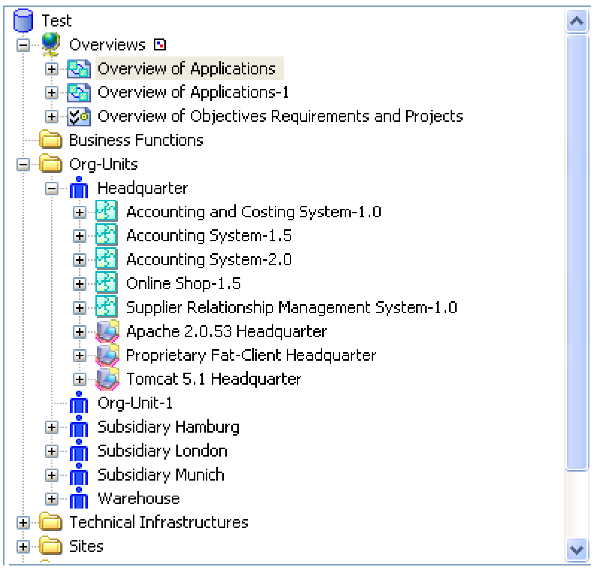


Figure 13.2.: *MEGA Modeling Suite* - axis Importing, Editing, and Validating: Tree-like navigator for repository content

Rating: 5

13.1.2. Creating Visualizations

MEGA Modeling Suite provides a number of different diagram types, which can be used for displaying information from the underlying repository. Nevertheless, most of these diagram types are *graphlayout maps* in terms of our definition (cf. Chapter C) - differentiated by the classes that can be displayed on them. The diagram types (and thus the selections of classes) are motivated from a methodological background, according to which they are sufficient for EA management. If the visualization to be created corresponds to one of these diagram types, the tool can provide strong support, while in other cases the user can adapt the method via a dialog. The graphlayout maps, see Figure 13.3, can be generated semi-automatically. Therefore, the objects to be displayed have to be dragged on the canvas, while the corresponding associations are automatically represented by connecting lines.

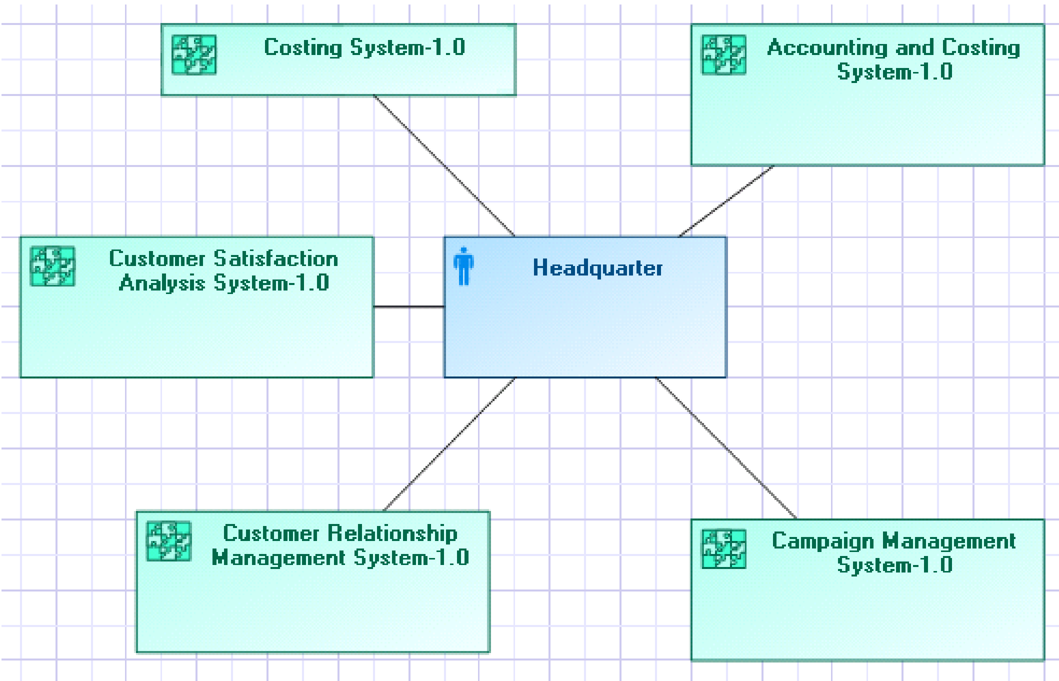


Figure 13.3.: *MEGA Modeling Suite* - axis Creating Visualizations: Semi-automatically generated graphlayout map

Making use of the visual flexibility provided by *MEGA Modeling Suite* visualizations resembling a *cluster map* (cf. Figure 13.4) can be created utilizing the *Overview of Applications* diagram. Nevertheless, the creation does not have strong automation, as e. g. the symbols representing application systems are positioned manually, although some relationship related semantics of symbol nesting exists (see Section 13.1.3). However, the linear symbols representing communication relationships between application systems are created automatically. In order to achieve full automation, the user could employ the built-in scripting capabilities of the tool.

A *process support map* displaying the business processes of the enteprise together with the application systems supporting them at different locations can be created in *MEGA Modeling Suite* employing some manual effort. First, the user has to adapt the template backing the *Overview of Business Processes* diagram to allow modeling of application systems. Second, the user can make use of the visual flexibility provided by the diagram editor to create a process support map as shown in Figure 13.5. Nevertheless, the relative positioning of the symbols representing application systems does not directly comprise a semantic relationship between the aligned objects in the underlying repository.

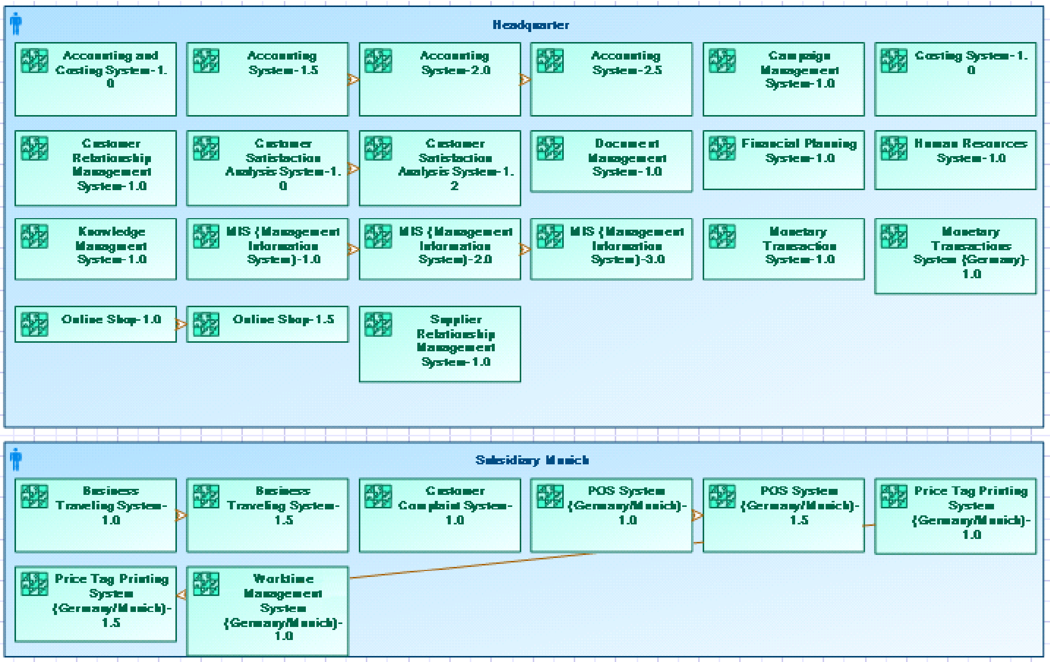


Figure 13.4.: *MEGA Modeling Suite* - axis Creating Visualizations: Manually created cluster map

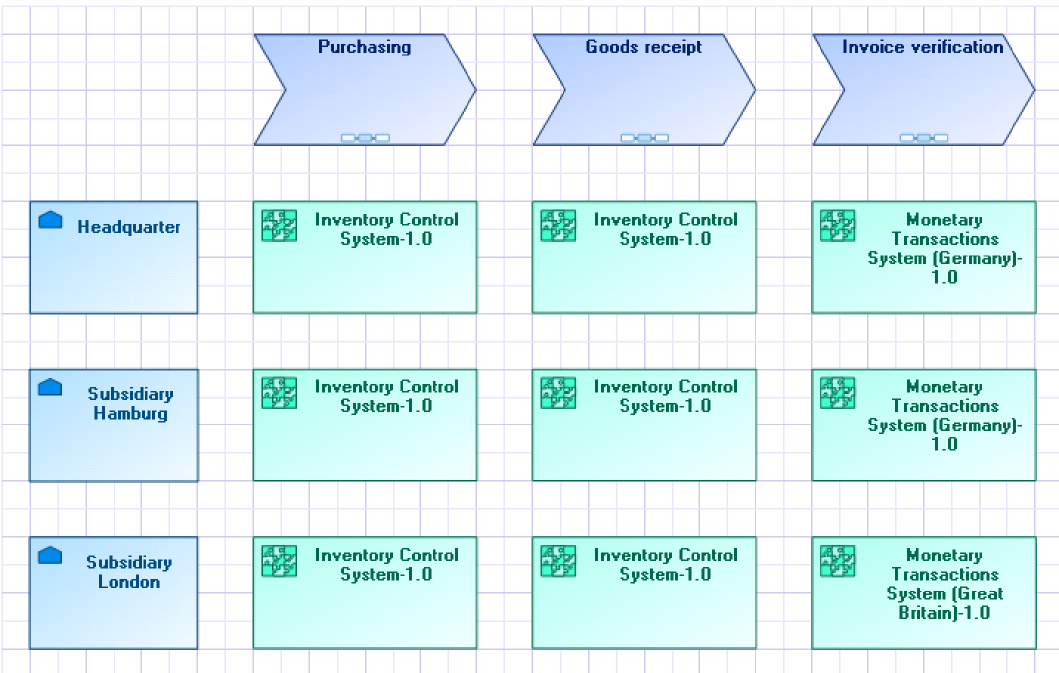


Figure 13.5.: *MEGA Modeling Suite* - axis Creating Visualizations: Manually created process support map

A *time interval map* displaying an overview on the lifecycle information of e. g. an application system cannot directly be created using the visualization techniques provided by *MEGA Modeling Suite*, but could be created utilizing the scripting capabilities present in the tool. Additionally, as temporal information (e. g. for lifecycles) can be stored in the information model, it is possible to execute an *analysis* on an existing model taking into account such temporal aspects, e. g. using highlighting. For more information on this see Section 13.1.3.

A *swimlane diagram*, e. g. displaying strategies, associated goals, and affected application system can be generated semi-automatically (cf. Figure 13.6) - with automation regarding the linear connections between the symbols, but without direct support for swimlanes containing instances only of a distinct class.

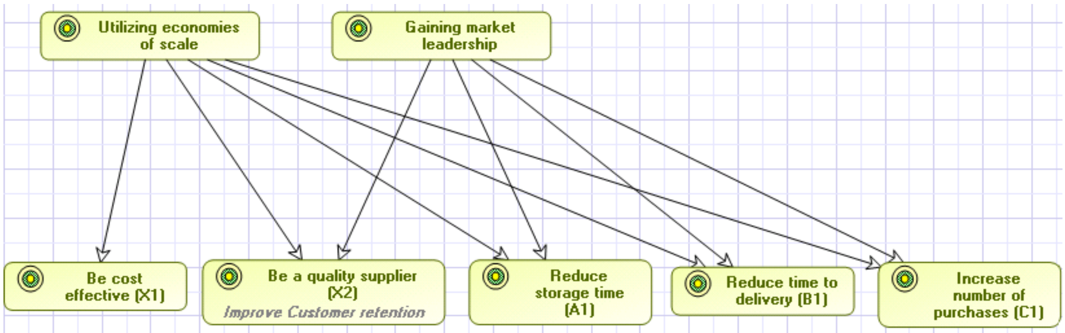


Figure 13.6.: *MEGA Modeling Suite* - axis Creating Visualizations: Manually created swimlane diagram

MEGA Modeling Suite does not provide direct support for *portfolio matrices*, but diagrams of that type can easily be created making use of the strong Microsoft Excel integration of the tool. Additionally, a skilled programmer could employ VBA macro programming to automatically export data to Microsoft Excel in order to create such a visualization.

Rating: 5

13.1.3. Interacting with, Editing of, and Annotating Visualizations

MEGA Modeling Suite provides a high flexibility regarding the adaptation of existing visualizations, especially concerning changes to the layout. The user can flexibly resize and move existing symbols, although a possibly existing semantics of the (relative) position of the symbol is not in all diagram types affected by the change to the diagram. Nevertheless, the *Overview of Applications* diagram partially supports the concept of *nesting* symbols to express semantic information from the underlying repository. Consequently, moving a nested symbol out of a symbol represented a related instance leads to the automatic creation of a linear symbol connecting both related symbols, see Figure 13.7. Nevertheless, the user can change the visualization without causing consequent changes on the underlying repository data. This is e. g. possible, as one can drag an application system from a surrounding organizational unit to another organizational unit, without the respective relationship being updated in the data. When changes have been applied to the repository data, which contrast the state displayed in the diagrams (e. g. as an application system is missing), the user is informed on that changes, when reopening the diagram. He then has the option to choose, whether the diagram should be adapted by adding symbols at the top left or removing not longer existing ones, or not.

Additionally, the user can select the symbol for representing an instance of a specific class from the set of symbol templates associated with the corresponding class. This set of templates can be extended and changed by the user, if neccessary. Symbols representing a given instance from the repository can be hidden in visualizations manually.

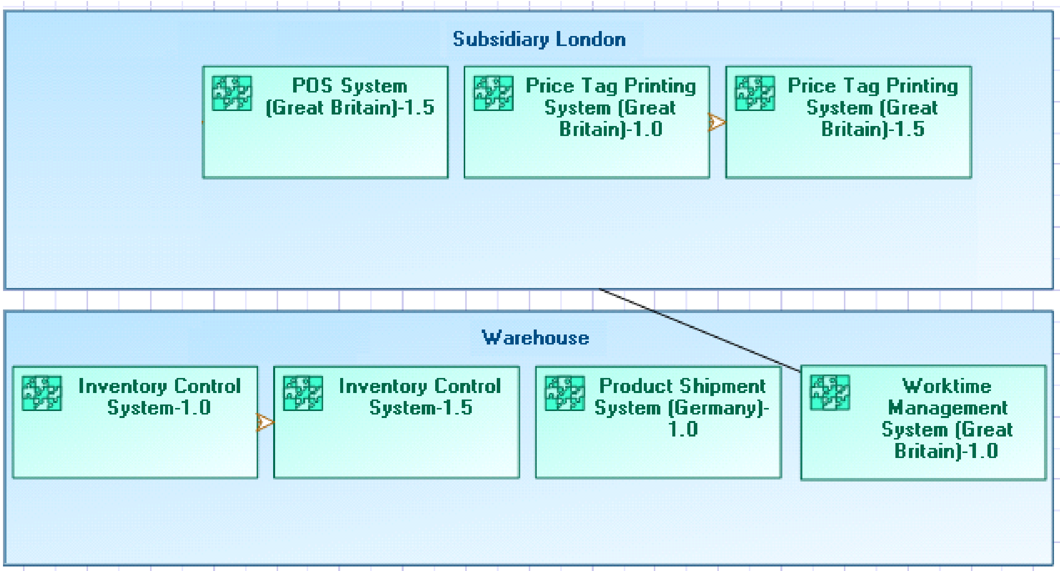


Figure 13.7.: *MEGA Modeling Suite* - axis Interacting with, Editing of, and Annotating Visualizations: Manually changed cluster map

Concerning the annotation of visualizations with additional information representing attribute values, *MEGA Modeling Suite* provides two different ways. The first one, which is to be considered the less flexible one, employs *computed shapes*. These shapes have been specially defined in the built-in shape editor to change their visual appearance in respect to an attribute value of the information object represented by them. If more sophisticated considerations should cause the appearance of the shape to change, scripting functionalities could be used. This is possible due to the direct association between the template of a symbol and an information model class. In addition to these semantically consistent ways for influencing the visual appearance of a symbol, the user can further employ the built-in shape editor for creating adapted templates of symbols (e. g. being colored in a different way) for supporting arbitrary changes to the visual make-up of a diagram. Finally, *MEGA Modeling Suite* provides another quite powerful way for creating an annotated visualization - the *analysis* mechanism.

An analysis in the terminology of *MEGA Modeling Suite* can be regarded as a set of automatically generated visualizations together with a *visual query*, which is subsequently applied to all existing visualizations containing at least one symbol representing an instance affected by the query. As the result of an analysis, the user is displayed the automatically generated visualizations, e. g. bar charts or kiviat diagrams. In addition, a list of visualizations already present in the repository is created with each of the visualizations graphically annotated in a way as specified in the query. Thereby, *MEGA Modeling Suite* is able to combine flexibility of visualization creation and adaptation on the one hand with powerful reporting and annotation mechanisms on the other hand, thus leveraging recognition of visualizations by the user. For creating such an analysis the user has to specify, which objects should be analyzed in which context and must provide a corresponding analysis report. For specifying the objects to be analyzed, the user can utilize a query assistance dialog, in which the filtering for the objects to be affected can be described. After having specified the query an analysis report can be created employing the assistance of dialogs. The actual analysis to be performed is defined using the *templating* mechanism as offered by *MEGA Modeling Suite*. In addition to the 75 analysis templates targeting a broad variety of EA management tasks, a trained user can define own analysis templates to be applied on the visualizations, therefore determining which graphical annotations should be created.

Rating: 4

13.1.4. Flexibility of the Information Model

MEGA Modeling Suite comes shipped with a comprehensive information model containing about 250 classes usable for modeling the enterprise architecture. Additionally, the information model can be adapted by the user via the *MetaStudio* module of *MEGA Modeling Suite*, as the model itself is realized as instance data in an object-oriented repository. Consequently, the user can create new classes and update user defined ones, e. g. concerning their name, as well as delete user defined classes extending the predefined information model. Concerning attributes the user can introduce new and strongly typed attributes as well as edit or delete existing user defined ones. The same is true for associations, which can be created, edited, or deleted, as long as user defined ones are concerned. By limiting the adaptations to the information model to user definable concepts, the basic model cannot be changed, thereby ensuring compatibility to future versions of the tool. In addition to the basic modeling concepts as alluded to above, the concept of an *attributed association* is supported. The classes, attributes, and associations are governed within the repository of *MEGA Modeling Suite*, which is said to be fully *MOF*-compliant[1]. Additionally, the repository provides support for so called *MetaAttributeGroups*, which can be used for grouping together closely related attributes.

In addition to the information modeling capabilities as outlined above, the repository of *MEGA Modeling Suite* offers more sophisticated functionalities, especially regarding the enforcement of constraints. This concept is realized in the repository by *ModelingRules*, which group together so called *MetaTest* instances. These instances are define as *macros* using the VBS language, which are actually executed in case of a change of the modeled data. Thereby, simple consistency checks as e. g. the enforcement of association multiplicities, but also more complex constraints e. g. affecting attribute inderdependence can be realize. Nevertheless, the user can limit access to this powerful constraint enforcement mechanisms to a special user group, having sufficient programming experience. By introducing constraints, it is possible to enforce and ensure model consistency especially in a distributed and collaborative modeling environment.

The concept of a predefined symbolic representation for a specific class is commonly used in *MEGA Modeling Suite*, with the possibility for the user to add further symbols or change existing ones by utilizing an external editor to adapt the symbol files in common graphic or icon formats. The current information model, especially concerning user made extensions to the basic information model of *MEGA Modeling Suite*, can manually be visualized in the MetaStudio module in an UML-like notation, as shown in Figure 13.8. These visualizations of the information model can be used for visually adapting the model, e. g. by introducing new classes or associations. The adapted information model of *MEGA Modeling Suite* can be exported to proprietary, *MEGA Modeling Suite*-specific formats.

Rating: 7

13.1.5. Communication and Collaboration Support

For leveraging the communication of EA management relevant information throughout the enterprise, *MEGA Modeling Suite* provides the built-in functionality of templates, which can be used to export data from the underlying repository to HTML-files and visualizations to images. While this functionality being the most lightweight way for accessing data from *MEGA Modeling Suite*, it is not the only one. If more sophisticated functionalities concerning communication and collaborative work are considered, the additional tool *MEGA Advisor*[2] can be used. Below, a quick but not comprehensive overview on the functionalities offered by the *MEGA Advisor* is given.

Using the *Advisor* tool, the user can utilize a web browser to access information from the underlying repository of the modeling suite. The data is thereby accessed on a real-time basis, while also additional

[1] *MOF* in this context refers to the OMG's *Meta Object Facility*.

[2] *MEGA Advisor* has not been evaluated in depth in the context of this survey.

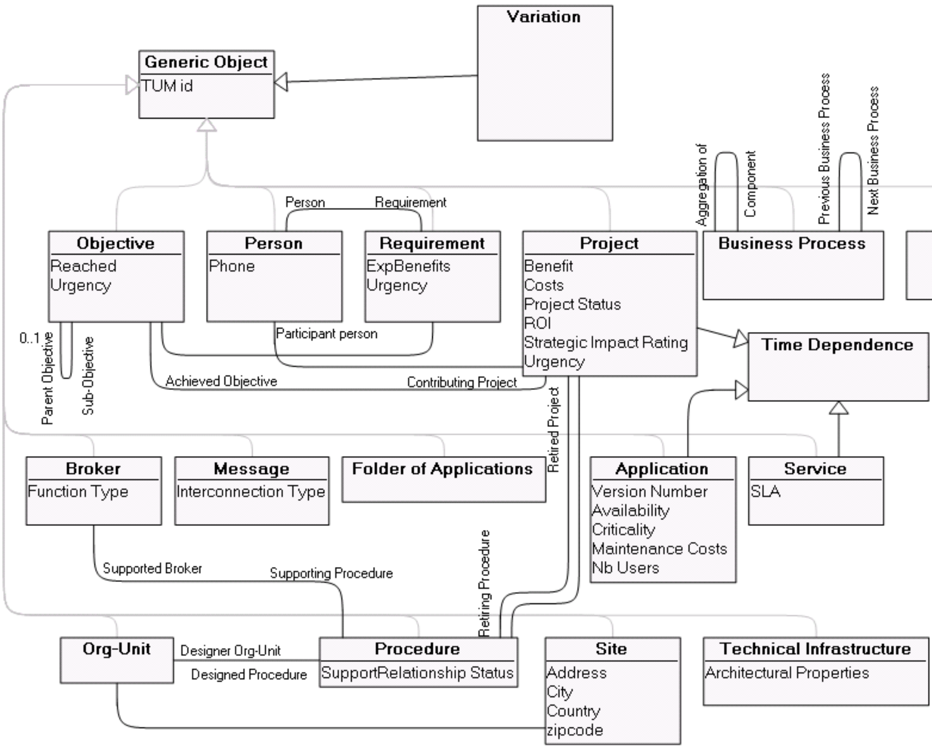


Figure 13.8.: *MEGA Modeling Suite* - axis Flexibility of the Information Model: Diagram displaying adapted parts of the MEGA information model

static pages can be included in the websites. The dynamically retrieved information is based on queries on the repository (cf. Sections 13.1.3 and 13.1.7 for more detailed information on the querying mechanisms).

The *Advisor* (cf. Figure 13.9) provides mechanisms leveraging convenient access to information, such as strong searching mechanisms. Nevertheless, as the *Advisor* is mainly a tool for communicating information, editing data in the repository via the Advisor's user interface is not directly possible, but could be enabled employing some customization by vendor.

Concerning role-based access to information from the repository (both in the rich client as well as in the Advisor), *MEGA Modeling Suite* provides the concept of user *profiles*. This access control mechanism employs *levels of confidentiality*, via which the user can be granted reading rights on different objects and visualizations[3]. The levels of confidentiality are complemented by the *authorizations* mechanism, which is used for restricting the overall editing capabiltities available in a certain profile.

Workflows for approving object changes are not natively supported but can be implemented using specialized customer defined attributes for objects. These attributes can subsequently be used in synchronizing objects between different data repositories, which represent different stages of approvement. In order to perform synchronization of these repositories the user can rely on the import and export capabilities of *MEGA Modeling Suite* as outlined in Section 13.1.1. The automated creation of notifications related to a change affecting specific objects is not directly supported.

When further concerning collaborative work, it has to be emphasized, that *MEGA Modeling Suite* provides support therefore via a multitude of concepts. In order to prevent simultaneous changes to

[3]These rights apply both in the rich client and in the web-based *Advisor*.

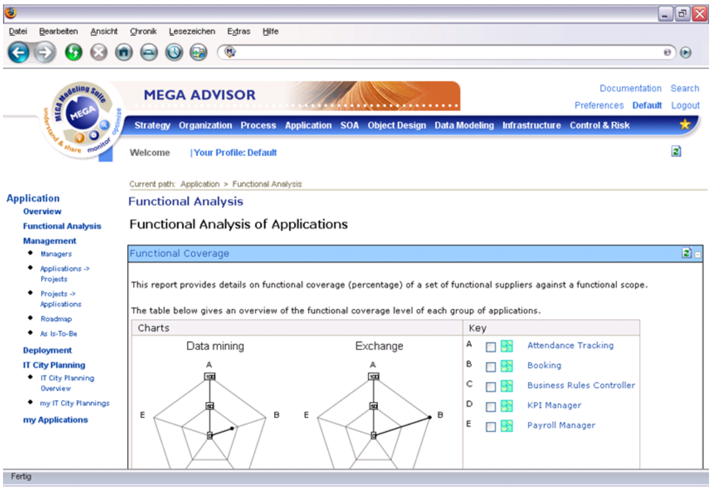


Figure 13.9.: *MEGA Modeling Suite* - axis Communication and Collaboration Support: Browsing the repository information and diagrams in the *MEGA Modeling Suite Advisor*

the same objects, the user implicitely performs a *check-out*, if the object is changed directly or using a visualization. When saving the changes a *check-in* is performed, actually commiting the updates to the underlying repository. If such a strict locking is not desired, the user can switch to optimistic locking, which, in case of concurrent changes, rises a collision report. The same report is used, if after utilizing the *offline-mode* of *MEGA Modeling Suite*, the information is re-synchronized with the central repository, of which a local copy has been created by switching to the detached mode.

Rating: 5

13.1.6. Support of large scale Data

MEGA Modeling Suite can import data on large scale application landscapes via the intergrated scripting capabilities as alluded to in Section 13.1.1 as well as via the Excel Editor plugin. The import of 1000, 5000, or even 10000 application systems can be imported in a reasonable time, although no "live" performance can be expected. As far as automated visualizations are supported (cf. Section 13.1.2) the data from the repository can be visualized though some minor performance diminishment is experienced, when the user wants to interact with the visualization, e. g. by resizing or moving a symbol. Storing the adapted visualization is executed quick, although no real-time performance is brought up.

In order to facilitate the handling of such large data sets about the application systems employed in an enterprise, *MEGA Modeling Suite* provides the concept of *City Planning Areas*. This concept can be used to group application systems to larger building blocks within the application landscape and further utilize these groupings in visualizations, e. g. to reduce the overall visualization complexity by reducing the number of lines displayed. Therein, the city planning area is handled as a single block to which the constituent application systems' interconnections can be aggregated.

Rating: 5

13.1.7. Impact Analysis and Reporting

Impact analysis computations are widely used and well supported within *MEGA Modeling Suite*. For the creation of such an analysis the user can rely on a dialog, see Figure 13.10, in which the user can select the class of objects to be analyzed. Subsequently, the user can define conditions on the attributes of the object, which shall be fulfilled (or not) in order to match the query. Also, relationships between objects can be traversed in querying, e. g. finding all the application systems related to a certain business process is possible by querying the applications and demanding an association to a business process with a given name. In order to hide the inherent complexity of queries on the repository from the untrained user, *MEGA Modeling Suite* offers the possibility to create and store *named* queries. These queries can be created by e. g. a skilled user and subsequently utilized by every users. These mechanism is especially powerful, as a stored query can define parameters, e. g. the business process, of which the supporting applications should be determined. In executing the query, the user can provide the needed value in a typesafe manner, thus leveraging flexibility of the querying mechanisms.

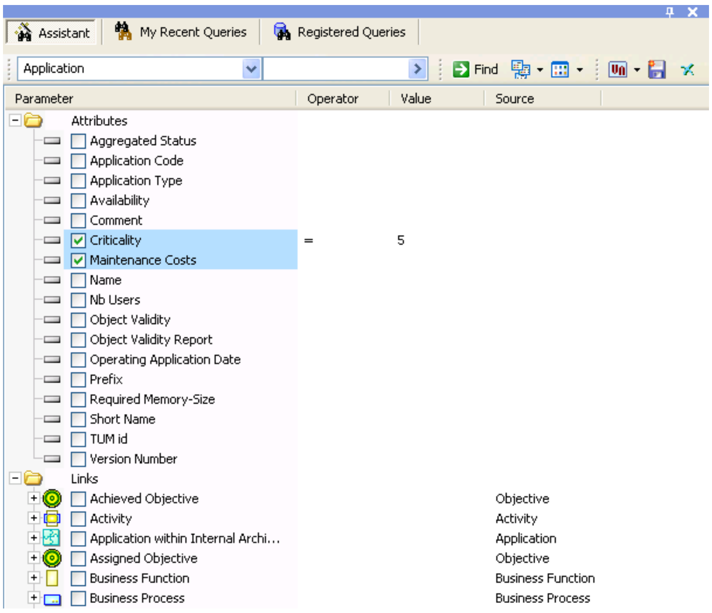


Figure 13.10.: *MEGA Modeling Suite* - axis Impact Analysis and Reporting: Dialog for creating user defined queries

As an alternative to utilizing the dialog for creating queries, the user can also specify them in a declarative language, which resembles the terms of standard SQL. This language is also the outcome of the query definition via the dialog.

Nevertheless, when calling for more complex computations as e. g. aggregation of values, these functionalities are not directly available in the querying wizard. Here, the scripting functionalities of *MEGA Modeling Suite* can provide the additional expressiveness needed for those more sophisticated impact analysis tasks.

As described above, the results of impact analysis computations can be reused throughout the tool for performing visual analysis on existing diagrams, as e. g. highlighting specific objects.

Rating: 6

13.1.8. Usability

MEGA Modeling Suite offers a well-structured user interface, mainly made up of a hierarchical navigator (cf. Figure 13.2) for the repository contents and a canvas for displaying diagrams or an overview (cf. Figure 13.11) of the possible analyses. Especially, this overview is quite helpful, when user defined analyses based on user defined queries are to be used or created. Additional windows, e. g. for defining queries, can be superimposed on the basic windows - hiding again, if they lose the focus except they have been *pinpointed*. A special welcome window, providing quick navigation to commonly used elements in the tool is further provided, especially useful for novice users.

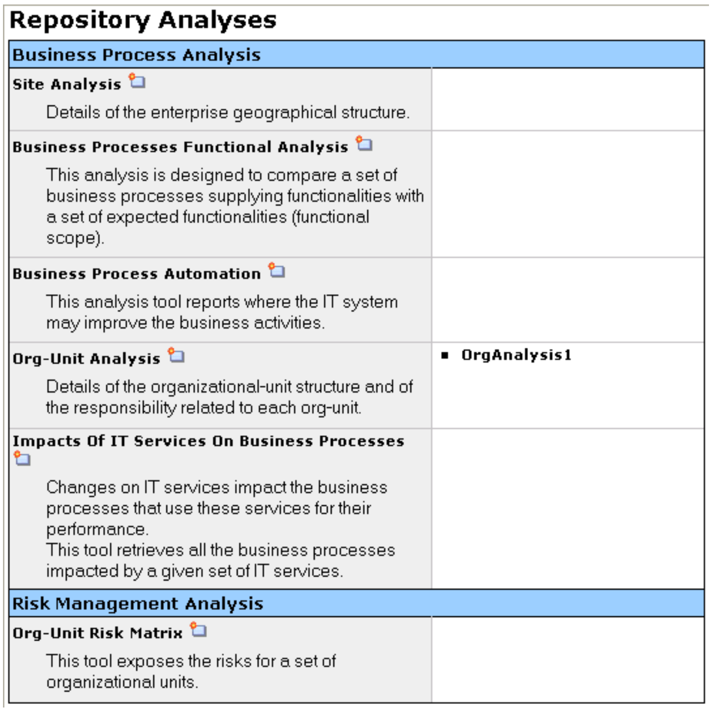


Figure 13.11.: *MEGA Modeling Suite* - axis Usability: Overview of the available analyses

The navigation between the different windows of *MEGA Modeling Suite* is easily performable, although a novice user might at first get lost in the multitude of options and views, from which he can chose. Nevertheless, mechanisms for efficient browsing as known from internet browser, namely a *back button* and a *browse history* can be useful to see interesting and previously used views at a glance.

For leveraging more sophisticaced concepts of *MEGA Modeling Suite* the user can rely on specific editors, such like an integrated scripting editor or a shape editor for changing the templates for visual representations of repository concepts. In addition, the integrated *MetaStudio* can be used for performing adaptations to the underlying information model in a convenient way. An additional component *MEGA Administration* can be used to graphically manage user roles and define the corresponding access restrictions.

Finally, *MEGA Modeling Suite* comes shipped with a comprehensive help system, additionally providing context sensitive information for the user. In addition, the help system likewise the user interface provides support for multiple languages.

Rating: 6

13.2. Evaluation of EA Management Support

The following Section describes the results of the scenarios simulation for EA management support.

13.2.1. Landscape Management

Central concept for modelling the evolution of the application landscape in *MEGA Modeling Suite* is the *Variation* concept. By this concept, the user can introduce planned[4] states for specific application systems (it would also be possible to create variations of other object, e. g. infrastructure elements, although this functionality has not been evaluated here). After having introduced planning variants, the user can employ the predefined *Varied Object Analysis* to get an overview on the evolution of the elements in the application landscape in a report as shown in Figure 13.12.

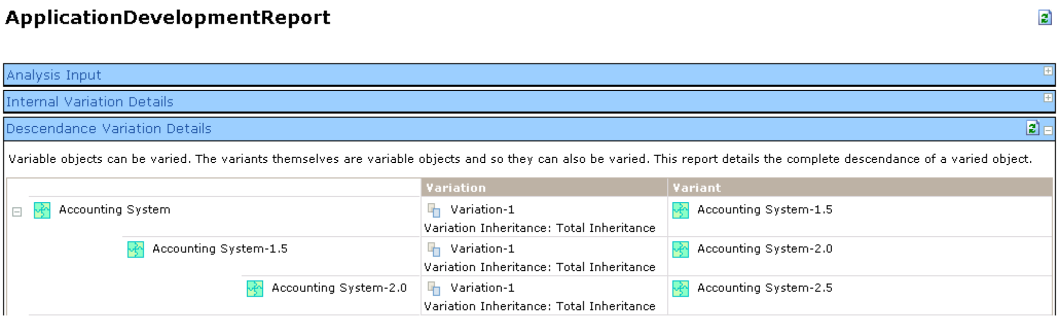


Figure 13.12.: *MEGA Modeling Suite* - axis Landscape Management: Analysis showing variations for an application system

The concept of *Variants* can be considered to be very powerful, allowing the user to select from different types of variations ranging from *complete inheritance* to *no inheritance*. Thereby, the user can designate for every variation of a source, which attribute values should be inherited from the source object. While the variants are used to maintain the relationships between the different versions of an application system, the user can additionally define for each variant a period of validity using so called *Time Periods*. Utilizing this concept, it is possible to run an *As-Is – To-Be Architecture Analysis* determining for a given point in time, which changes between the current and the planned architecture of the application landscape will have occured. These changes can subsequently be used for annotating an existing visualization with color-coding, see Figure 13.13. Using this analysis it is possible to perform comparisons between the current and any selected future or past landscape, or an arbitrary landscape scenario[5]. Additionally, variants of the application systems are always edited *in place*, therefore it is not directly possible to access a previous planning state. Nevertheless, archive repositories could be used to access this information.

Concerning the overall capabilities of *MEGA Modeling Suite* regarding landscape management, one important fact has to be noted. The variants of application systems can be introduced and maintained unrelated to projects, which may have to be executed in order to perform the actual transformation between the variants. This can be seen as an important benefit, considering the degrees of freedom in creating planned versions of the application landscape. Nevertheless, if projects should be modeled as the drivers of transformation between variants of the application systems, they can be associated manually, by relating the validity periods of the application systems and the different periods (phases) in project execution.

[4]Throughout *MEGA Modeling Suite* the term *As-Is* is used for the current and *To-Be* for planned versions of an element.

[5]A scenario herein refers to a planning scenario as introduced in Section 4.2.1.

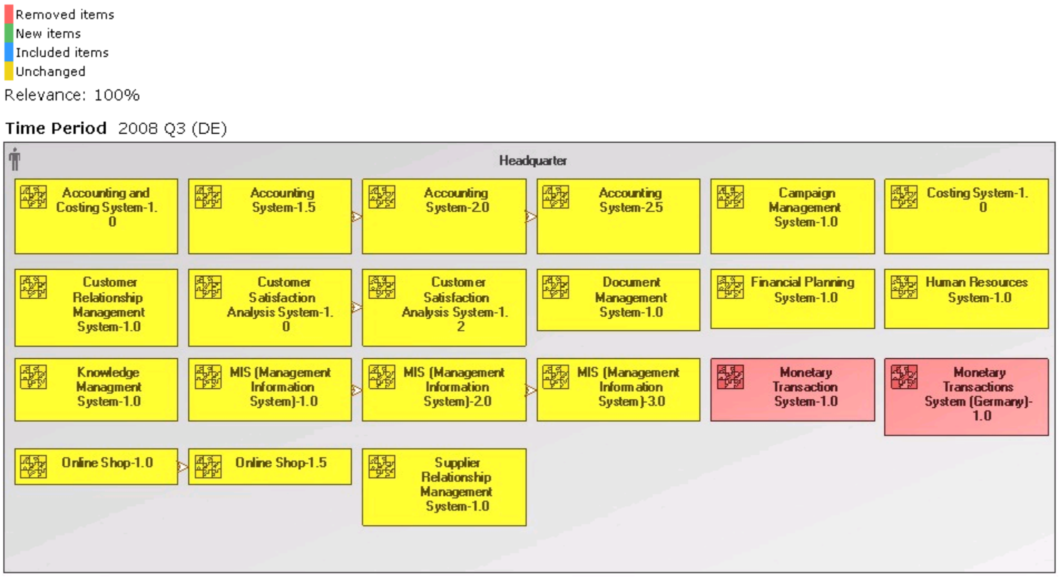


Figure 13.13.: *MEGA Modeling Suite* - axis Landscape Management: Analysis showing variations for an application system

The procedure consistency is mostly given. The concept of *Variants* allows to manage the evolution of application systems on a very fine grained level. Nevertheless, tracking historic planning states is not directly supported.

The procedure integration is completely given. *MEGA Modeling Suite* can make use of data previously entered in the tool, especially in defining plans for future application landscapes utilizing the powerful variant mechanism.

Rating: 5

13.2.2. Demand Management

For supporting *Demands* as introduced in the information model, *MEGA Modeling Suite* provides two different concepts: *Requirements* and *Functionalities*. While the first concept merely represents a sort of general demand, which can be applied to application systems as well as to processes, projects, or goals, the second concept explicitely targets application systems representing a *functional requirement*[6]. As a consequence, the first concept has a spectrum of possible uses broader as demanded in the scenario, causing some degrees of freedom in modeling, which might not be desirable under all circumstances. The second concept, whilst strongly focused on application systems, may also in some cases not be the fully suitable, as demands could target non-functional requirements, as e. g. response time, too. These non functional requirements then had to be represented by a *functionality* object, which might somewhat cause confusion. Consequent, it is deemed advisable to take a decision for every single demand to be represented in *MEGA Modeling Suite* in order to find the more appropriate type, the demand can be represented with.

For displaying, which demands affect which application systems, the user can adapt the *Overview of Objectives and Projects* diagram by allowing the visualization of application systems. Thereby, a visualization of the demand to application system relationships as shown in Figure 13.14 can be created manually.

[6]The term *functional requirement* is taken from the field of software engineering, see for example [So01].

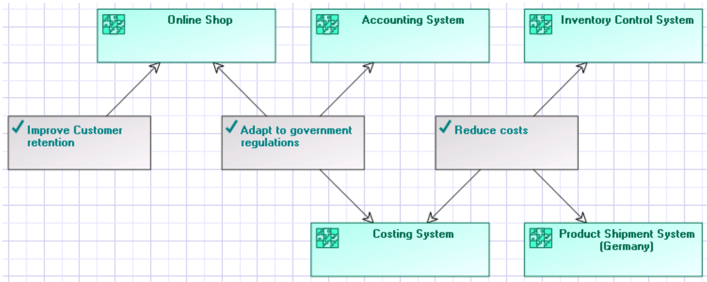


Figure 13.14.: *MEGA Modeling Suite* - axis Demand Management: Diagram showing the relationship between demands and application systems

Displaying similar information in a tabular report is directly possible making use of a matrix like visualization of relationships.

The procedure consistency is mostly given. *MEGA Modeling Suite* does not directly support the concept of demands such that an existing concept has to be reused or the methodology has to be adapted.

The procedure integration is completely given. *MEGA Modeling Suite* can reuse data previously entered into the tool.

Rating: 5

13.2.3. Project Portfolio Management

Projects can be documented in the repository of the platform and employed in different reports and analysis. For decomposing projects, *MEGA Modeling Suite* offers the concept of subprojects. Both, subprojects and projects can be associated to affected elements of the EA, as e. g. application systems, business processes, etc. (cf. Figure 13.16). Nevertheless, as *MEGA Modeling Suite* does not provide a concept completely matching the *SupportRelationship* (see *SoCaStore* information model in Section 3.2), it is not directly possible for projects just to e. g. affect the business process support provided by an application system at a distinct organizational unit. This information must explicitely taken into account, when running analyses using the predefined analysis templates, as the association used in *MEGA Modeling Suite* to model the support information does not provide means for explicating temporal dependency[7].

Measures for project management, as e. g. ROI or project costs can be maintained in *MEGA Modeling Suite* and exported to tabular reports as well as to Microsoft Excel files, by which the user can easily create portfolio matrices similar to the ones displayed in Section 4.2.3. Furthermore, an evaluation concerning the number of projects affecting a specific application system, the can be performed by the user by relying on the predefined *Project Management Analysis*, via which a visualization as shown in Figure 13.15 can be created.

The tabular listing of the analysis report as shown in Figure 13.15 can be used to determine project interdependencies caused by affected application systems. Thereby, the predefined report is limited to the impact of projects on application systems or to the affected business processes, while impacts to organizational units can as alluded to above due to the absence of the *SupportRelationship*[8] not be display directly.

[7] According to *MEGA*, a function for modeling temporal aspects of associations will be included in an upcoming support release in 2008.

[8] The association between business process and application system can be attributed to hold an organizational unit property. Nevertheless, the visualization of an association as line can only employ two ends.

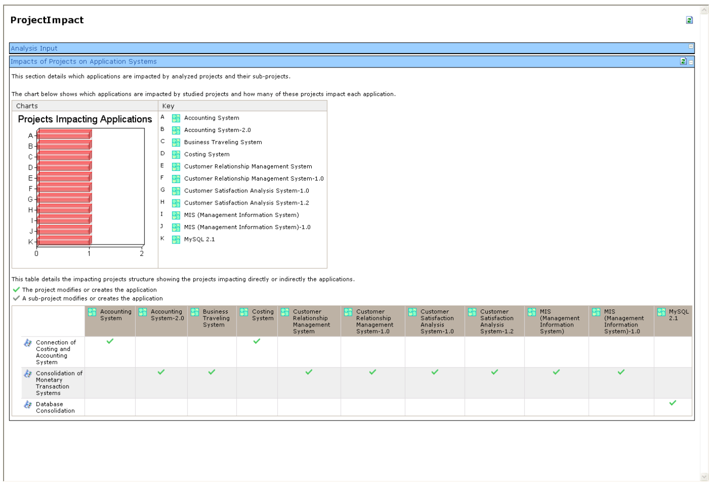


Figure 13.15.: *MEGA Modeling Suite* - axis Project Portfolio Management: Dialog showing, which project affects which architecture element

Nevertheless, as outlined in Section 13.2.1, the different states of the application landscape are not directly connected to the changing projects but e. g. through *Time Periods* used for expressing the time for which the specific concept (e. g. project or application system) can be considered valid. An analysis for determining project interdependencies to the planned states of the application landscape can be performed using the predefined *Project Analysis* template.

The procedure consistency is mostly given. *MEGA Modeling Suite* performs well in supporting management of project impact on application systems, although especially deriving transitively linked dependencies is not supported without customization.

The procedure integration is mostly given. *MEGA Modeling Suite* can reuse data previously entered into the tool, although information about variants created in landscape management cannot directly be used as input for project management.

Rating: 5

13.2.4. Synchronization Management

Synchronization issues concerning multiple projects currently affecting the application landscape can partially be addressed using *MEGA Modeling Suite*. The project schedule can be modeled relying on the *Time Period* concept, which allows the user to supply absolute start and end dates for the validity of an object. Instances of this concept can be created and assigned to the application systems affected by a specific project as well as to the project itself. The first assignment is necessary, as the predefined *Time Constraint Analysis* does not directly account for transitive constraints; nevertheless, such a functionality could be implemented via customizing the corresponding analysis template. In order to facilitate assignment of time periods to application systems affected by the selected project, the user can either use the *Project Analysis Template* (cf. Figure 13.15) or employ a dialog showing the associated project deliverables, see Figure 13.16.

Utilizing time periods, it is also possible to explicate project interdependencies, which e. g. arise from application systems affected by different projects. As alluded to above, a transitive calculation of these dependencies is not directly supported. Nevertheless, the functionality can be leveraged employing some manual work by the user, i. e. querying the application systems affected by more than one project and subsequently establishing temporal constraints between the time periods of the

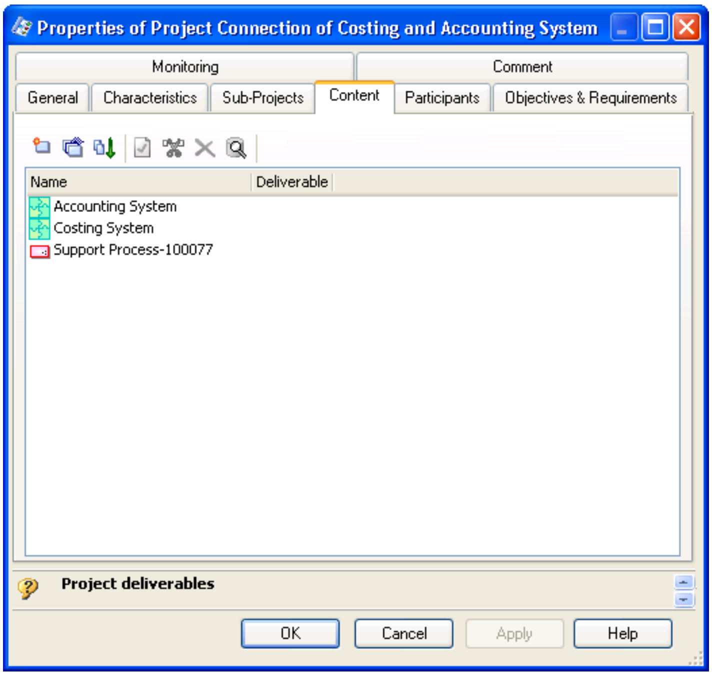


Figure 13.16.: *MEGA Modeling Suite* - axis Synchronization Management: Dialog showing the architecture elements affected by a project

corresponding projects. This approach additionally allows the user to control time related project interdependencies on a fine grained level.

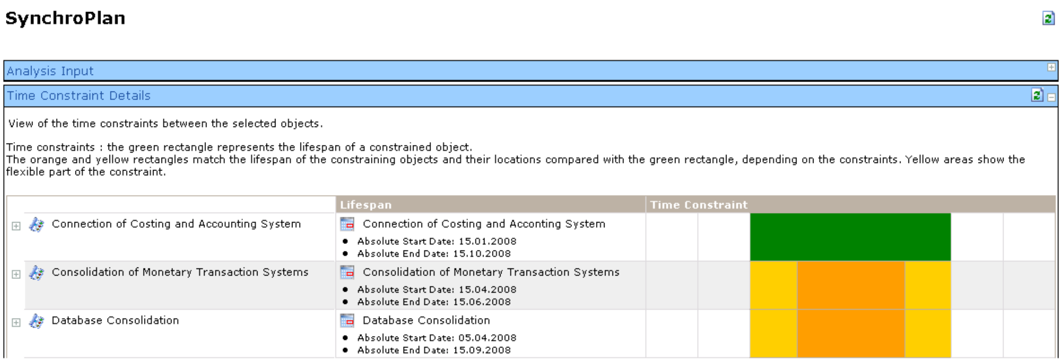


Figure 13.17.: *MEGA Modeling Suite* - axis Synchronization Management: Analysis showing temporal project dependencies

Having maintained the information as alluded to above, the user can make use of the predefined *Time Constraint Analysis* of *MEGA Modeling Suite* to create a synchronization plan as shown in Figure 13.17. Such a visualization can also be utilized in displaying the impact of a project delay, although this concept is not directly supported in *MEGA Modeling Suite*. Nevertheless, the user can adapt the schedule of the project in the corresponding time period, so the analysis as described above can be run again. The same approach can be utilized, if the time of validity for an application system is changed, e. g. by supplying a designated time for retirement - again the *Time Constraint Analysis* can provide neccessary information.

The procedure consistency is mostly given. The concept of *Time periods* allows a fine grained management of synchronization issues, although delays are not directly supported.

The procedure integration is completely given. *MEGA Modeling Suite* can reuse data especially from project portfolio management and from landscape management.

Rating: 6

13.2.5. Strategies and Goals Management

Strategies and goals can be mapped to the standard information model of *MEGA Modeling Suite* as *Objectives*. For the relationship between, the *cause and effect* relationship can be utilized. Although this mapping provides the neccessary functionality for facilitating strategies and goals management, it has to be noted, that both concept are mapped to the same class from the *MEGA Modeling Suite* information model. For differentiating these concepts, the user can utilize a predefined attribute, indicating, whether the goal is *strategic* or *operative*. Based on this information, a modeling constraint could be introduced, for ensuring that strategic goals are used as strategies and therefore not connected to projects or demands. Subsequently, a swim-lane diagram (see Figure 13.18) displaying, which strategies are transitively connected to a given application system can be created, although some manual querying is needed.

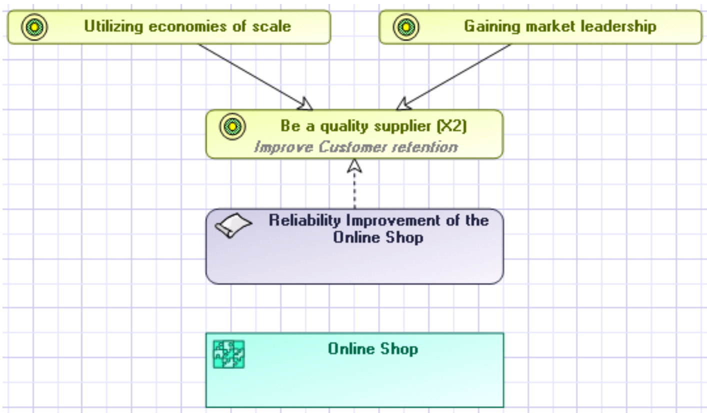


Figure 13.18.: *MEGA Modeling Suite* - axis Strategies and Goals Management: Diagram showing, which strategies are related to a given application system element

Traversing from the strategies to the (transitively) affected objects is also supported in *MEGA Modeling Suite*, although the same manual querying interactions are required to model a diagram, like the one shown in Figure 13.19. Nevertheless, via the integrated *Dialog* functionality a visualization of the impact can be created (cf. Figure 13.20). The same is possible, customizing an analysis template automizing the creation of such visualizations to the point, where the connected elements would be highlighted.

If more sophisticated calculations concerning goal achievment are required, *MEGA Modeling Suite* does not provide direct support, although these computations could be performed employing scripting mechanisms within *MEGA Modeling Suite* to remotely control Microsoft Excel spreadsheet calculations. While this way requires a user having some experience in using the scripting mechanisms, a low-key alternative is provided utilizing the integrated exporting facilities to hand the data over to Microsoft Excel.

The procedure consistency is mostly given. The cause-and-effect relationship between objectives allows to govern strategies and goals, although analyses employ some manual effort in querying.

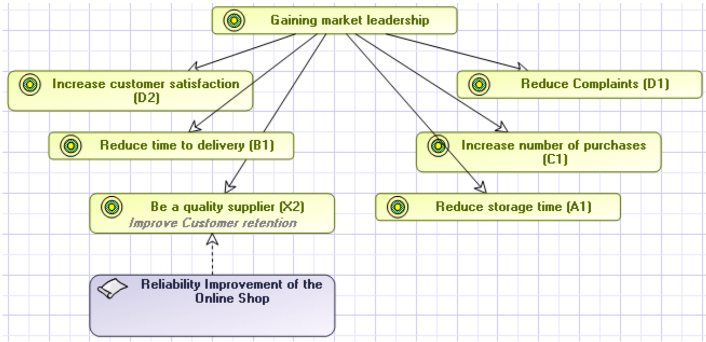


Figure 13.19.: *MEGA Modeling Suite* - axis Project Portfolio Management: Diagram showing, which goals are affected by a given strategy

The procedure integration is completely given. *MEGA Modeling Suite* can reuse data especially from landscape and project portfolio management.

Rating: 5

13.2.6. Business Object Management

Business objects can be managed using the *Message* concept and its related content provided by *MEGA Modeling Suite*. A *Message* is considered according to its definition to be a particular (control) flow between a sender and a receiver object, which may convey information. The content description for the message can be used to define the type of information exchanged, if any - which applies to e. g. money exchange. In this notion, the *Message content* can be seen as option to model the *Business Object* as introduced in the information model (see Section 3.2). This is especially true, as different messages can share the same message content, which provides a consistent option to model the exchange of the same business object in the communication of different application systems. Nevertheless, messages and their contents have a broader meaning in *MEGA Modeling Suite* than demanded from the information model. A message is not limited to be exchanged between application systems, but can also be exchanged between business processes or organizational units. Therefore, this notion of message can be used to consistently model flows spanning different levels of the EA, e. g. from processes downwards to actual application systems. Additionally, the content of a message needs not to represent an information or data exchange, but can also be used to model the exchange of material or finance goods. Nevertheless, utilizing the capabilities to manually create diagrams, it is possible to create a visualization of the business object exchange as the one shown in Figure 13.21. In addition to the functionality required in the scenario, it is possible to add information about the actual infrastructure elements employed in message exchange.

As messages provide a means, by which information exchange can be modeled, they are not intended to model the operations performed on the objects contained therein. Therefore, information describing which application system performs which operation on which business object during the execution of a specific business process, cannot be governed within *MEGA Modeling Suite*. Nevertheless, *MEGA Modeling Suite* offers strong capabilities helpful in modeling the actual buildup of a business object. These structural aspects of business object types, as e. g. attributes or relationships can be entered and maintained using the data modeler provided within *MEGA Modeling Suite*. Such a data model can be visualized employing some manual work, leading to a diagram as shown in Figure 13.22.

The procedure consistency is mostly given. While business objects and therefore operations executed upon them are not directly supported, communication and control flows can be modeled in *MEGA Modeling Suite*.

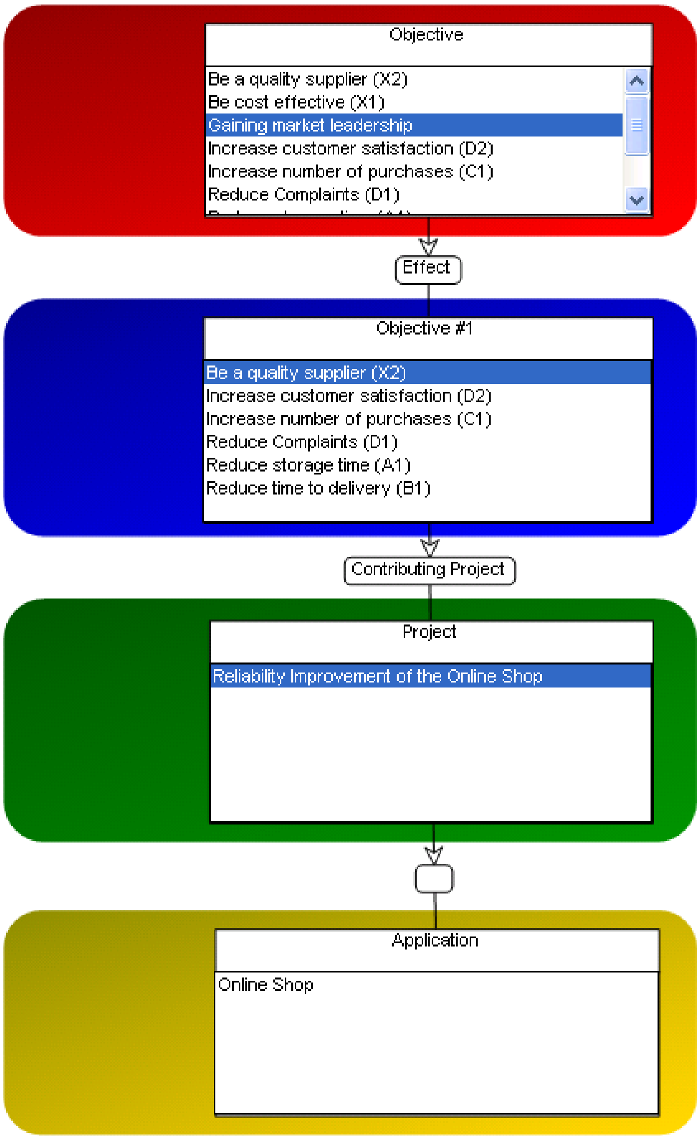


Figure 13.20.: *MEGA Modeling Suite* - axis Project Portfolio Management: Diagram showing the relationships between strategy, goal, project, and application system

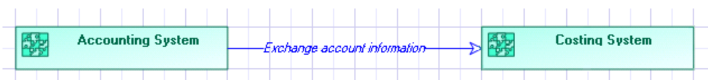


Figure 13.21.: *MEGA Modeling Suite* - axis Business Object Management: Diagram showing, application systems exchange which business objects

The procedure integration is completely given. *MEGA Modeling Suite* can reuse data previously entered in landscape management.

Rating: 5

Figure 13.22.: *MEGA Modeling Suite* - axis Business Object Management: Diagram showing the internal data structure of a business object

13.2.7. SOA Transformation

The information model of *MEGA Modeling Suite* does not directly support all concepts, which are employed in the SOA transformation approach as described in Section 4.2.7. This is especially true, when concerning the business objects and the operations performed on them, which cannot directly be modeled in *MEGA Modeling Suite* (cf. Section 13.2.6), but are the starting point for the hybrid SOA transformation approach as analyzed in this scenario. As a consequence, an approach to analyze the EA from a SOA perspective is subsequently evaluated. Nevertheless, as alluded to above, via customization of the information model and the analyses templates, the deliverables of this scenario (cf. Section 13.2.6) could be created with *MEGA Modeling Suite*.

Services in the terminology of *MEGA Modeling Suite* can be introduced as intermediary concepts used for connecting the procedures, business processes consist of, to the applications, which actually realize this procedures. Services of this type are called *IS Services* in *MEGA Modeling Suite*. If further decomposition of the architecture in a service oriented way is needed, *MEGA Modeling Suite* also provides the concept of the *IT Service*, of which multiple ones can contribute to the realization of an IS Service. Adhering to this modeling paradigm, the services on the different levels can be included in interaction scenarios detailing on how they provide actual business support.

The procedure consistency is mostly given. As outlined above, business objects are not directly supported in *MEGA Modeling Suite*. They could be used in a SOA transformation approach, if introduced by customization.

The procedure integration is mostly given. *MEGA Modeling Suite* can reuse data previously entered, except for detailled information concerning business objects.

Rating: 5

13.2.8. IT Architecture Management

In the information model as provided by *MEGA Modeling Suite* no concepts for describing abstract architectural principles for an application system are provided[9]. Nevertheless, the actual architectural building blocks can be reflected by two different concepts from the information model. On the one hand, the user can utilize the UML support of *MEGA Modeling Suite* in order to model *components*[10] (see Figure 13.23). On the other hand, the core *MEGA Modeling Suite* information model provides a *sub-applications* relationship between different application systems. This relationship could also be used in modeling the internal structure of an application. These both mechanisms are fairly quite disjoint and do not provide an automated exchange. As a consequence, the user would have to duplicate information, when wanting to leverage all of *MEGA Modeling Suite*'s application architecture

[9]According to *MEGA* an upcoming update (to be released in 2008) is going to include an information model adaptation supporting additional architecture concepts, suitable in the context of this scenario.

[10]*Components* in this context are defined as in the UML superstructure [OMG05].

management capabilities.

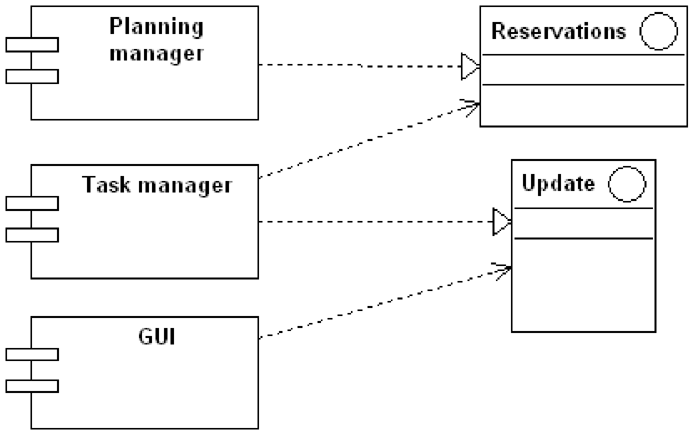


Figure 13.23.: *MEGA Modeling Suite* - axis IT Architecture Management: Diagram the internal component architecture of an application system

The strict separation of the two modeling alternatives might cause slight inconveniences, when making use of the powerful analysis templates as provided by *MEGA Modeling Suite* out of the box. No template does exist taking into account the comprehensive architecture of an application system, as the modeling alternatives alluded to above exist fairly disconnected.

For representing the more abstract concepts of application architecture such as *abstract technologies* or *architectural blueprints MEGA Modeling Suite* does not provide direct support, although some concepts as e. g. databases are represented by specific classes in the information model. For leveraging full-scale architectural blueprinting as described in Section 4.2.8, the user has to introduce additional concepts (for adaptability of the information model see Section 13.1.4) and to create additional analysis templates. Thereby, reports and graphically annotated deliverables as demanded in this scenario similar to the one displayed in Figure 13.13 could be created by performing an analysis.

Finally, it would also be possible to reuse the concept of the *Technical Infrastructure* to represent abstract architectural concepts, so that predefined analysis templates could be reused. Nevertheless, doing so would result in a situation, where architectural concepts on different levels of abstraction would be represented by a single class in the information model. While an additional attribute could be introduced for distinguishing and could be used in defining specific queries, the data modeled in accordance to that information model might cause slight confusion for an occasional user.

The procedure consistency is mostly given. The information model of *MEGA Modeling Suite* has to be adapted and new analysis templates have to be defined to perform application architecture management.

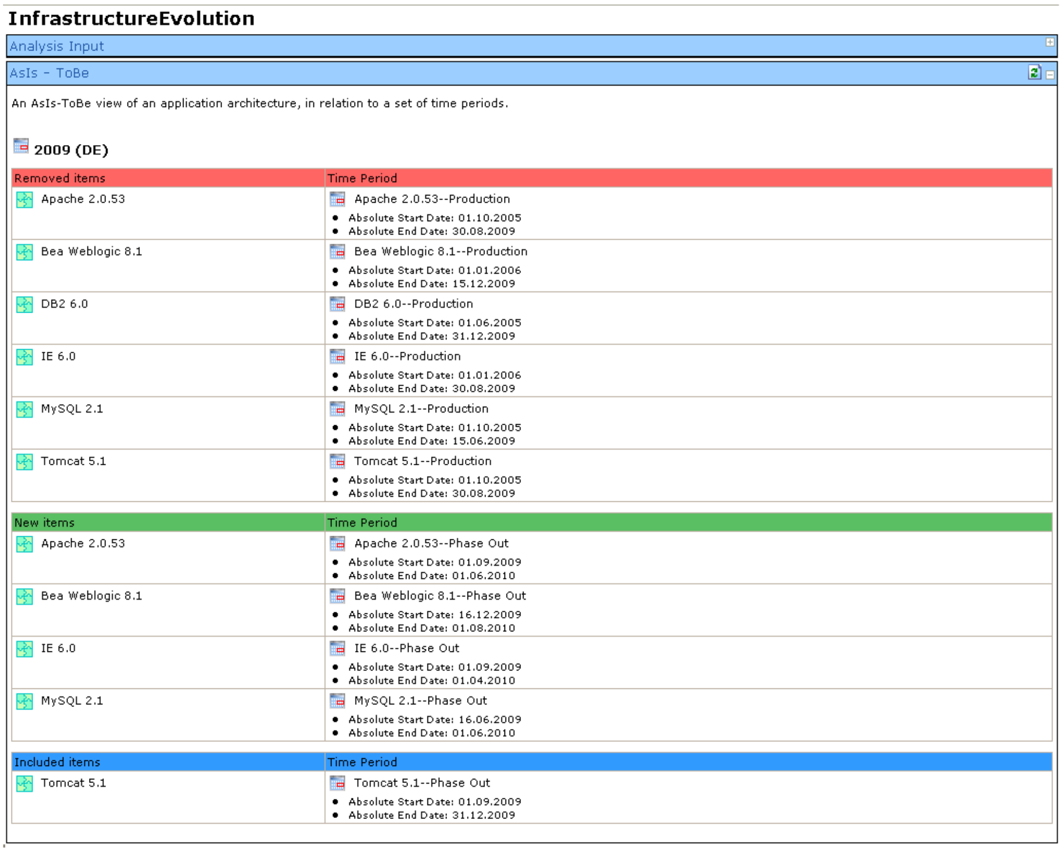
The procedure integration is mostly given. *MEGA Modeling Suite* can reuse data previously entered into the tool, while the out-of-the-box support for application architecture management is not completely integrated with itself.

Rating: 5

13.2.9. Infrastructure Management

The information model of *MEGA Modeling Suite* provides no general *infrastructure component* class, but specialized classes for distinguished types of infrastructure, e. g. *Database* or *Technical Infrastructure*. The description below strongly focuses on the usage of the database class, but similar considerations would be true for the other infrastructure components, for which a corresponding class in *MEGA Modeling Suite* exists. In case none such class could be found, the user can make use of the possibilities to adapt the information model as described in Section 13.1.4 at the cost that the predefined analysis templates do not directly support the newly introduced classes.

A database object in *MEGA Modeling Suite* does not directly provide attributes for maintaining lifecycle information. While for other infrastructure elemtents, the user could attach a *Time Period* (cf. Section 13.2.1) for expressing validity - no similar concept is available for databases. As a consequence, the predefined validity analysis cannot be directly reused here. Nevertheless, the user can employ the report by representing a database also by an application system, for which the predefined report can be leveraged. Thereby, a visualization showing, which technological infrastructure (database) runs out of support at a given point in time can be created (cf. Figure 13.24).


InfrastructureEvolution

Analysis Input

AsIs - ToBe

An AsIs-ToBe view of an application architecture, in relation to a set of time periods.

2009 (DE)

Removed items	Time Period
Apache 2.0.53	Apache 2.0.53--Production • Absolute Start Date: 01.10.2005 • Absolute End Date: 30.08.2009
Bea Weblogic 8.1	Bea Weblogic 8.1--Production • Absolute Start Date: 01.01.2006 • Absolute End Date: 15.12.2009
DB2 6.0	DB2 6.0--Production • Absolute Start Date: 01.06.2005 • Absolute End Date: 31.12.2009
IE 6.0	IE 6.0--Production • Absolute Start Date: 01.01.2006 • Absolute End Date: 30.08.2009
MySQL 2.1	MySQL 2.1--Production • Absolute Start Date: 01.10.2005 • Absolute End Date: 15.06.2009
Tomcat 5.1	Tomcat 5.1--Production • Absolute Start Date: 01.10.2005 • Absolute End Date: 30.08.2009

New items	Time Period
Apache 2.0.53	Apache 2.0.53--Phase Out • Absolute Start Date: 01.09.2009 • Absolute End Date: 01.06.2010
Bea Weblogic 8.1	Bea Weblogic 8.1--Phase Out • Absolute Start Date: 16.12.2009 • Absolute End Date: 01.08.2010
IE 6.0	IE 6.0--Phase Out • Absolute Start Date: 01.09.2009 • Absolute End Date: 01.04.2010
MySQL 2.1	MySQL 2.1--Phase Out • Absolute Start Date: 16.06.2009 • Absolute End Date: 01.06.2010

Included items	Time Period
Tomcat 5.1	Tomcat 5.1--Phase Out • Absolute Start Date: 01.09.2009 • Absolute End Date: 31.12.2009

Figure 13.24.: *MEGA Modeling Suite* - axis Infrastructure Management: Report showing which technological infrastructure is changed in 2009

Not taking into consideration the lifecycle information for databases, but focusing on the dependency of an application system on a database, a predefined analysis template called *infrastructure compliance* can be used. Therein, the user can determine, which databases are used by which application systems. Further, the number of total usages of a given database technology can be computed, giving an anticipation of the impact, which arises if the given technology runs out of support. Finally, the user can distinguish databases in *expected*, *accepted*, and *prohibited* ones. Therein, the user can set databases running out of support to be prohibited, getting an overview on their usage throughout the application landscape. A visualization created via this infrastructure compliance analysis is shown in Figure 13.25.

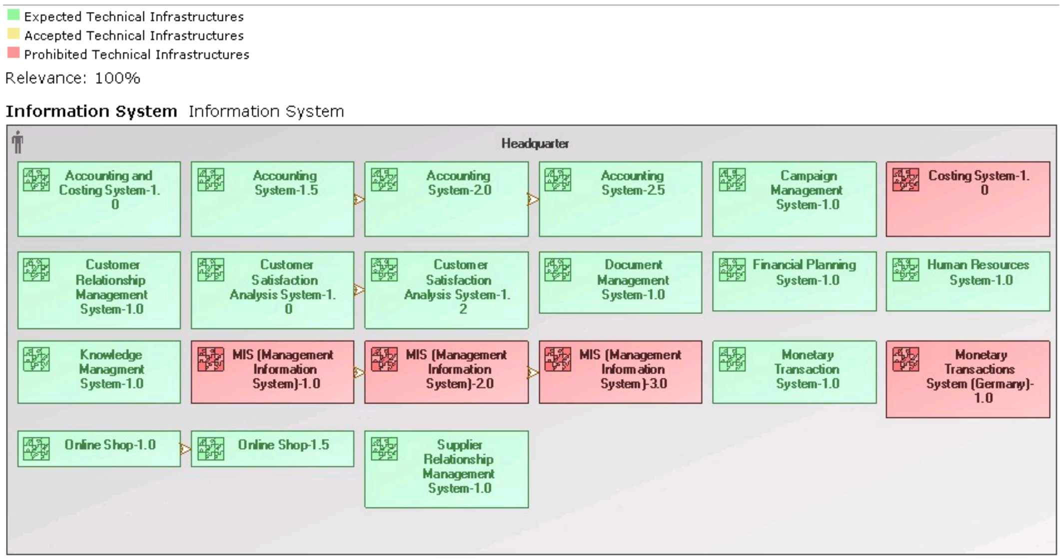


Figure 13.25.: *MEGA Modeling Suite* - axis Infrastructure Management: Annotated visualization of the application landscape indicating the usage of a prohibited database system

The procedure consistency is mostly given. The information model of *MEGA Modeling Suite* does not directly support lifecycle information for databases.

The procedure integration is completely given. *MEGA Modeling Suite* can reuse data previously entered into the tool.

Rating: 6

13.3. Tool Vendor's Profile

Rolf Irion, Managing Director

MEGA International

Founded in 1991, headquartered in Paris and with offices and representation around the world, MEGA International is a global leader in enterprise architecture, offering solutions to improve company performance and ensure regulatory compliance. The company is an active contributing member to the Object Management Group (OMG). MEGA solutions provide executives with the ability to understand, control and govern information at the enterprise level through its prominent modelling software, proven best practice methods, and expert consulting in business process analysis, IT alignment, operational risk management, internal control and multi-regulatory compliance.

Company Profile

MEGA International is recognized as a leading provider of solutions for Enterprise Architecture, Business Process Analysis, IT Architecture, Operational Risk, Internal Control and multi-regulatory Compliance. This market position relies on three pillars that have underpinned the company's vision, development and strategy since its creation:

- Provide leading-edge management tools to support company executives' decisions
- Provide the right methods to analyze and solve complex issues
- Understand clients' and partners' requirements

MEGA's customers include Fortune 1000 companies in nearly any industry sector. MEGA's mission is to help clients improve their business performance through three complementary solutions. These allow MEGA's customers to **understand** the organization and assets of the enterprise (the system), **control** the mechanisms of value creation (business processes and operations) and to **govern** the alignment with the company's strategy, objectives and policies.

Solutions

MEGA centres its activities around three solution areas that cover a wide scope of concern: Business Process Analysis & Performance, IT Architecture & Governance, Regulation, Risk & Control. MEGA solutions are powered by three software tools.

MEGA Modeling Suite's main purpose is to build and reuse a consistent, powerful enterprise-wide repository covering the description of the business processes, human and IT resources and organization that support and implement the enterprise strategy. It embeds a powerful and proven methodology, and automates the publication of value-added deliverables that can be circulated across all stakeholders depending on their needs and profiles: a complete enterprise blueprint.

Because each company is specific, MEGA Modeling Suite is highly customizable while supporting commonly adopted standards, process improvement methods and modeling and design standards.

MEGA Process and MEGA Simulation support business managers in their discovery, description and improvement of the company's business processes. MEGA Architecture helps IT architects inventory and manage the enterprise IT assets and plan IT changes and the roadmap according to business changing requirements. MEGA Designer is the solution of choice for component, data, database and integration modeling. MEGA Control & Risk helps managers define risk management and control policies.

MEGA Exchange, MEGA Supervisor, MEGA Studio and MEGA Publisher support the integration of the platform within the customer's environment, the customization of the platform as well as the automated generation of value-added deliverables like websites and documentation.

Finally, Accelerators bootstrap customers in areas where they do not expect to start from a blank page.

MEGA's GRC strategy, based on the acquisition of a largely used technology for managing banking transactions, allows customers to benefit from the specific requirements stemming from transaction management: ability to manage heavy loads with excellent performance, guarantee of robustness and security, and ability to manage users located in different back-ends depending on their role and profile.

MEGA GRC Suite addresses the requirements of Operational Risks Management and supports the related day-to-day operations. It helps companies to comply with external or company-specific or industry-specific regulations. MEGA GRC Suite is a module and platform-based solution that helps define the risk management policy and procedures and streamline the control of their performance.

Communication, to and from stakeholders across the organization, is a key success factor of Enterprise Architecture, Business Process Improvement, and Governance, Risk & Compliance programs. **MEGA Advisor**'s dynamic web portal technology provides all stakeholders, including occasional users, LOB managers and decision makers, the right combination of models and near-real-time data and dashboards to back decision making. Dynamic access is established to MEGA Modeling Suite, MEGA GRC Suite and to third party systems data (e. g. Business Activity Monitoring).

Consulting & Services

As a leading *independent* provider of Enterprise Architecture methodologies, MEGA has developed a unique expertise in resolving complex business and IT challenges using modeling techniques. MEGA brings to its customer a successful combination of tools, methodology and extensive project experience.

MEGA Consulting provides training, coaching and implementation services ranging from methodology adoption and customization to undertaking the entirety of our customer's Enterprise Architecture, Business Process Analysis and Governance or Risk and Compliance projects.

MEGA provides its customers with a high-level consulting team of more than 150 consultants specialized in EA and excellent Business Processes and all related modeling techniques. This is one of the largest teams in the world focusing exclusively on EA and Process excellence, from strategy, governance to the support of all execution phases.

MEGA University's training department mission is to train MEGA's customers and partners both on the methodology and the tools.

CHAPTER 14

Metastorm (ProVision)

Contents

ProVision provides an EA management tool with a focal point on modeling. Thereby, different frameworks and modeling languages are supported out-of-the-box, e. g. *DoDAF* or *Zachmann*, which can be switched back and forth by the user. The version of the tool evaluated in this survey is *ProVision* 6.0.

14.1. Evaluation of Specific Functionality

Below the results of simulating the scenarios from Section 4.1 with *ProVision* are explicated. Prior to the detailed description, the build-up of the tool is shorthanded. A central concept in *ProVision* is the *Repository*, which represents a container for one or more *Notebooks*. Each of these notebooks, which we will refer to as repository, contains its own objects and models. Leveraging the functionality of different notebooks, planning variants can be designed without corrupting the actual data (see Section 14.2.1 for more detail). Within each notebook, so-called *Snapshots* can be created, which can be used to freeze the current status of a notebook.

14.1.1. Importing, Editing, and Validating

ProVision supports several ways to import data from different formats, amongst other XMI for instance data or Microsoft Visio for diagrams. Importing data from Microsoft Excel files can be performed by mapping the worksheets to classes or links of the predefined information model and by mapping each column to a predefined attribute of the specified class. To import the customer specific parts of the information model, a template specifying the structure of the Microsoft Excel file should be exported first, filled with the respective data, and imported finally. Data transformation during the importing process is not supported directly by *ProVision*, but can be easily performed leveraging the functionalities provided by Microsoft Excel.

Incorrect data, e. g. data in a wrong format or of incorrect data types are not rejected during the import process, as consistency checks are not supported by *ProVision*. Missing ends for an imported link are generated automatically by *ProVision*, if the respective elements are not contained within the repository. After importing, a *completeness check* can be performed by *ProVision*, which checks missing values and returns a summary of all violations as shown in Figure 14.1. Thereby, the completeness check can be run on selected models but not on the repository itself.

The user can easily access data using the tree like navigator provided by the *Object Inventory* view, where the attributes and relations of an element are displayed. Editing an element can be performed by selecting it in the navigator view and choosing "detail" in the context menu, as double-clicking is not supported. A dialog can subsequently be used to edit the attribute values and relations of an object. Multiple simultaneous changes on data are not directly supported by *ProVision*, but can be easily performed by exporting the data to Microsoft Excel, carrying out the changes there, and reimporting the data into *ProVision*. Thereby, no conflict resolution is typically performed during the import process. However, a locking mechanism is provided by *ProVision*, which allows the user to check-out data change it and check it in later, ensuring that no changes made in the meantime will get overwritten.

Rating: 4

14.1.2. Creating Visualizations

In *ProVision* every visualization is created using a *Modeler* and specifying a *Subject*. While the Modeler defines the type of the visualization, including the elements which can be displayed, the

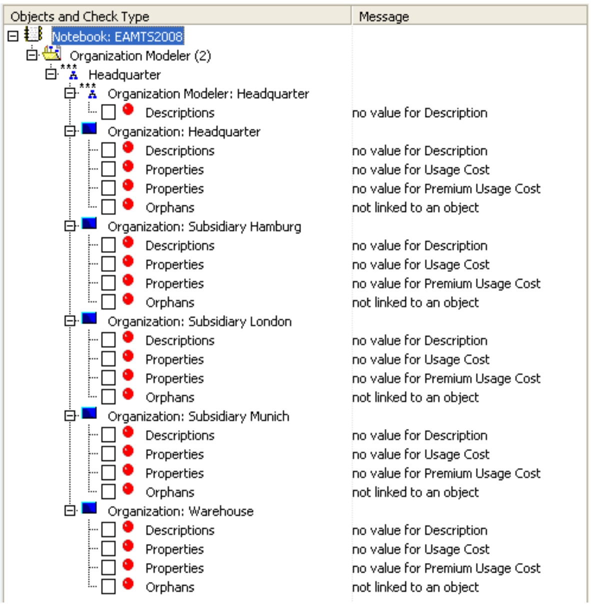


Figure 14.1.: *ProVision* - axis Importing, Editing, and Validating: List of all violations found by *completeness check*

Subject determines the base class of the visualization, which is used for navigating between different visualizations.

ProVision supports the creation of visualizations by providing drag and drop functionalities and simple pre-defined layouting mechanism, e. g. *hierarchy top down, hierarchy left right, fork top down, fork left right,* and *network*). A prerequisite for this automated layout is a link between the different objects that should be visualized. Further functionality concerning the creation of visualizations is provided by *ProVision*, e. g. automatic alignment, rotation, flipping, or ordering of objects on the canvas. Besides, for every object displayed on the diagram, *ProVision* provides a button to automatically include directly connected objects (neighbors) of the selected one, which can be chosen from a pick list. If the displayed object is a base class of another diagram, another button is displayed, which allows the navigation to the respective diagram.

Utilizing the drag and drop functionalities provided by *ProVision*, every visualization as introduced by the scenario as described in Section 4.1.2 can be created manually. An automated generation of the visualizations is not directly supported. Nevertheless, some of the required visualizations (e. g. time interval and portfolio matrices) can be generated leveraging the built-in functionality of *Crystal Reports*.

Due to the fact that the relative position of an element to others does not have a semantic meaning within *ProVision*, no automated generation of the process support map is possible. Nevertheless, leveraging the manual drawing possibilities provided by *ProVision* a cluster map (see e. g. Figure 14.2) as well as a process support map as visualized in Figure 14.8 can be created.

Swimlane diagrams, as shown in Figure 14.3, can be created semi-automatically using the neighbors functionality of *ProVision*. Thereby, traversing over more than one link (e. g. from *Project* over *SupportRelationship* to *DeployedApplicationSystem*) without displaying the traversed element is not possible. Nevertheless, traversing over elements in a visualization can be simulated leveraging the drag and drop functionality and drawing a *link*, which are connections between two objects without a semantic meaning. Thus, it must be noted, that the created visualization cannot be updated automatically, if changes in the underlying repository occur.

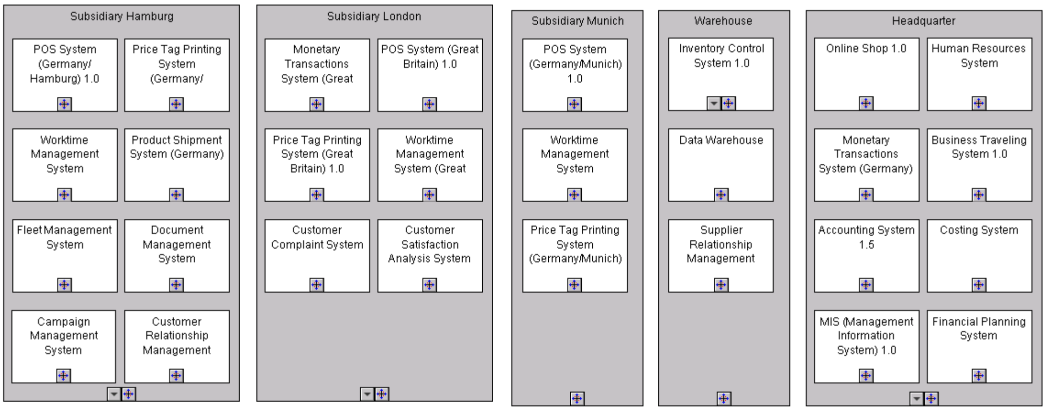


Figure 14.2.: *ProVision* - axis Creating Visualizations: Cluster map

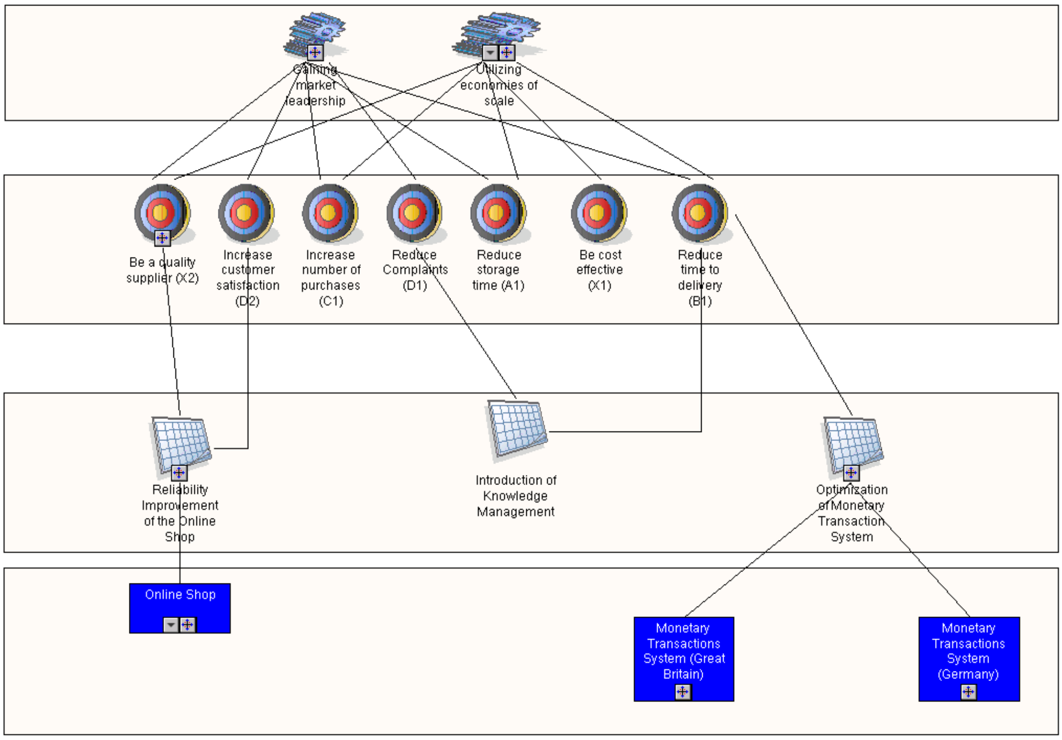


Figure 14.3.: *ProVision* - axis Creating Visualizations: Swimlane diagram

Leveraging the neighbor functionality provided by *ProVision*, graphlayout maps can be created almost automatically (see Figure 14.4). Thereby, the same limitations concerning the traversing functionality as mentioned above are given. Whereas, the creation of time interval maps is not supported by *ProVision*, these visualizations can be generated using the functionalities provided by *Crystal Reports* or the built-in drawing functionalities of *ProVision* (see Section 14.2.4 for examples).

Time interval maps can be created by drawing the necessary elements on a model, but the information in the chart has to be taken out of the repository by the user (see Figure 14.4). Another option to create time interval maps is to use *Crystal Reports*. The same ways it is possible to create portfolio matrices.

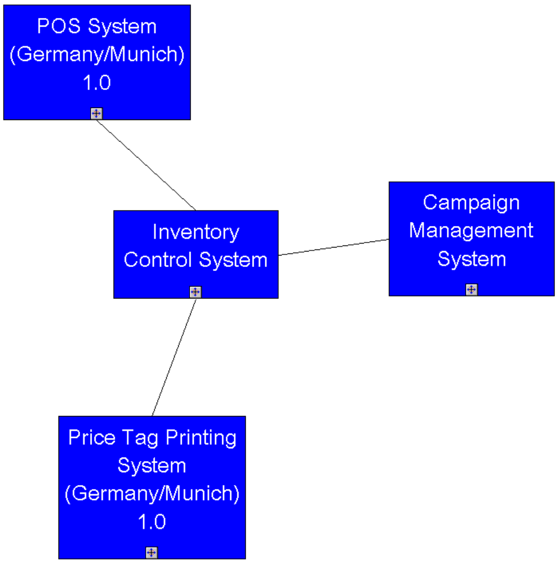


Figure 14.4.: *ProVision* - axis Creating Visualizations: Graphlayout map

Summarizing the capabilities of *ProVision* in respect to create visualizations it must be noted, that the tool focuses on providing flexible drawing possibilities. This approach delivers the possibility to create all the visualizations mentioned above on the expense of drawing them manually. Thereby, the visualized objects, which represent information from the repository, are linked to the respective object and updated automatically, if changes occur.

Rating: 3

14.1.3. Interacting with, Editing of, and Annotating Visualizations

ProVision provides different ways to interact with a visualization, e. g. double clicking on a displayed object allows the user to edit the underlying data using a dialog. Multiple changes concerning the visualization itself are possible, e. g. changing the style or size of a symbol. Furthermore, *ProVision* provides the possibility to propagate the changes concerning the style of a symbol representing an object from the repository to all other occurrences of the object. Every change performed on a visualization will be saved and considered the next time the model is opened.

ProVision differs between changes that are accomplished on the layout of a visualization and changes that are performed on the displayed data. Moving objects on the pane does not perform any changes in the underlying repository, as the position of an object on the pane does not posses a semantic meaning within *ProVision*. In contrast changes on the displayed data, e. g. the name of an object or creating a new object or link, will be reflected in the underlying repository.

The modeler provided by *ProVision* supports the use of layers, which offer the possibility to create frozen layers with fixed objects (unchangeable in layout and positioning) and layers with flexible objects. Using this layering mechanism, a basic support for highlighting or adding annotations is provided. Thereby, if a visualization should be annotated with symbols representing a certain aspect, the creator of the visualization has to look up the respective attribute and adapt the visualization by hand[1]. Furthermore, the layering principle can be used to hide or show selected elements. An example

[1]According to Metastorm the automatic adaption of an object style according to its attribute values is on the roadmap of development.

visualization with annotated aspects, that has been created manually, can be found in Section 14.2.8. Taking this manual annotation into consideration, no limitations concerning the possibility to annotate a visualization are given.

Further interaction capabilities, e. g. filtering a visualization or automated highlighting of objects of interest are not supported by *ProVision*. However, leveraging the neighbor functionality provided for every object displayed on a visualization, directly connected objects can be automatically visualized facilitating impact analyses. In addition, *ProVision* offers navigation buttons to ease navigation to related visualizations, which for example detail the inner structure of the respective object.

The creation of custom reports is not supported directly by *ProVision*, but can easily be achieved using the interface provided for *Crystal Reports* and its built-in functionality. The thereby created report definition files can be executed directly using *ProVision*.

Rating: 2

14.1.4. Communication and Collaboration Support

For supporting the communication and maintenance of EA management relevant information throughout the enterprise, *ProVision* provides the *KnowledgeExchange* server. This tool operates as a central server to manage users and access to databases. In addition, the tool provides a lightweight access to the information and models stored within the repository. Thereby, the data is accessed on a real-time basis. Whereas, limited interactions on models are supported, e. g. zooming, hiding layers, navigation between models, the model itself cannot be edited. Nevertheless, selecting an object by double-clicking on the respective symbol is supported as well as the editing of information using dialogs. The changes performed within the web-access can be committed to the repository. Furthermore, bookmarking is supported by the *KnowledgeExchange* server to provide quick navigation mechanisms.

In addition, *ProVision* support the publishing of models by exporting them as HTML files. Thereby, the makeup of the generated HTML files is similarly build as *ProVision* itself, containing a navigator on the left hand side and a main window displaying the selected elements. Whereas, the amount and steps of zoom levels can be configured during the publishing process, navigating utilizing visualizations is not directly supported. The export creates static HTML pages with images for the visualizations, therefore no direct access control is possible, but can be created using the functionality of a webserver. Similarly, editing and interacting with the visualizations or working with layers is not directly supported.

In order to avoid conflicts, if multiple users are working on the same data, the *KnowledgeExchange* server provides functionalities to check-out and lock repositories, notebooks, or objects. Thereby, a user can check-out the parts of a model, which should be worked on and later check-in the respective parts, without overwriting data changed in the meantime. Thereby, the status (locked or free) is propagated to other users utilizing a lock-symbol on the respective objects within the *Object Inventory* view and on the corresponding symbols used on diagrams.

Concerning role-based access to the information stored within the repository, the *KnowledgeExchange* server comes shipped with a built-in user and access management support. Users can be grouped into user groups, which can be modeled as hierarchy. The accessibility and visibility of elements contained within the repository can be defined on server, repository, notebook, object, or filter level. Thereby, a filter could e. g. be used to mark certain elements within the repository as *public* and apply a filter, which allows all users, which do no belong to a group, to view only elements defined as *public*.

Auditing and tracking of events is supported by the *KnowledgeExchange* server. The editing history can be viewed, which supports multiple filters, e. g. on users, objects, and events. A similar tracking of events, e. g. logon, logoff, check-out, check in, or creation of an object, is supported directly by *ProVision*, where a history is provided for an element (see Figure 14.5) containing information as creation or last modify date. Mechanisms for notifying users about changes on objects are not directly supported within the *KnowledgeExchange* server or *ProVision*. Nevertheless, within *ProVision*, it is possible to specify a time for an alert for a selected object. This functionality could be used to remember a user to check the actuality of the stored information.

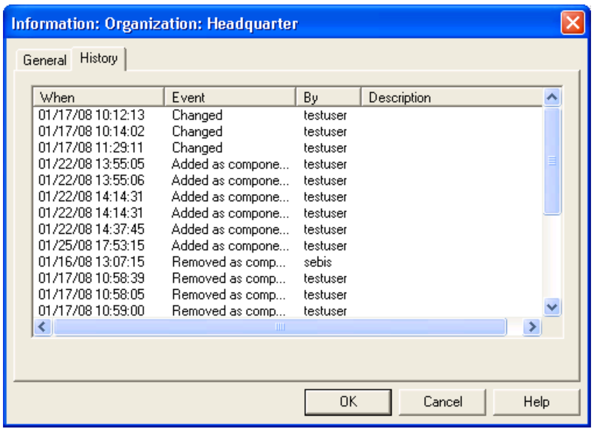


Figure 14.5.: *ProVision* - axis Communication and Collaboration Support: Event history

Rating: 3

14.1.5. Flexibility of the Information Model

ProVision comes shipped with predefined information models which are called *Modeling Languages*. These modeling languages implementing frameworks, e. g. Zachman, TOGAF, DODAF, can be extended using the 150 predefined so-called *Custom Objects*, which refer to classes in the terminology used through the survey. Thereby, these *Custom Objects* can be renamed and properties can be defined for them. Whereas datatypes, e. g. Integer, Boolean, Character, Date, Currency String, URL, can be defined for properties, default values or setting a property as mandatory is not supported. Custom types for properties can be defined by specifying a base type, e. g. Integer, and several constraints, e. g. minimum and maximum values. Leveraging this approach, enumerations can easily be defined. Further, typed arrays for a selected data type can be defined.

Relationships between object types can be defined in *ProVision* using so-called *Custom links*. Thereby, a link between two object types represents a bidirectionally m:n-relationship. Other multiplicities, e. g. 0...1, 1...*, 1...2, are not directly supported. Nevertheless, properties can be specified for a custom link type.

For every modeling language, it is possible to create several *Model Types*, which represent the visualizations, grids, or charts that can be created and define the elements, which can be visualized on these diagrams. These model types can be customized by changing the style and representation. The possibility to change the context object types - the elements, which can be displayed on the respective visualization - is only given on a restricted subset of model types.

Whereas predefined object types, attributes, and relationships cannot be deleted, custom objects can be changed or deleted according to the requirements of the user. If an existing element e. g. a custom object is deleted, the corresponding data is also removed from the repository. Unlike, the corresponding data sets, the visual representations of an empty custom object will remain within the visualizations, but lose its meaning and connection to the data contained within the repository.

ProVision offers some supplemental classes, e. g. *Artifacts, Cost Elements, Notes,* or *Qualifications*. These objects have a predefined meaning and are designed for special purposes: *Artifacts* can be used to add further information, e. g. Microsoft Word documents or URLs, to an object; *Cost Elements* can be used to define different types of costs, e. g. maintenance costs, which can be connected to projects for example; *Notes* can be used to e. g. short comments to an object; *Qualifications* have a predefined association to persons and can be used to define the qualification of the employee.

Furthermore, *ProVision* supports the concept of *Dimensions* to specify closely related objects. This concept can be used to e. g. define a dimension for versioned object types and non-versioned object types. Therefore, a dimension and its possible values, e. g. version numbers, have to be created. After the creation, a *projection* of objects can be specified, for instance application systems. Thus, an application system can exist within different dimensions, e. g. a versioned one and a non-versioned one. Leveraging this concept, these two objects can have different associations and attribute values.

Rating: 4

14.1.6. Support of large scale Data

ProVision can handle large scale data concerning the application landscape, containing thousands of application systems and their interconnections. Whereas, accessing information contained in the repository is not visibly affected by the amount of data, the import has to be performed in separate steps. Thereby, the large data set has to be split up in smaller parts and each part has to be imported separately. Nevertheless, after importing the data into *ProVision*, most actions, e. g. creating and editing of elements or attribute values is executed considerably fast.

Creating visualizations of the application landscape containing all application systems and their interconnections can be done manually. Thus, the elements can be visualized by leveraging the drag and drop functionality as described in Section 14.1.2. Nevertheless, as mentioned above, the tool's support to automatically create symbols representing selected elements is limited. Therefore, dragging and dropping elements from the inventory to the canvas has to be done in several steps, if a large amount of elements should be visualized. Interactions with such a map, e. g. resizing or moving a symbol, are executed considerably fast. Contrastingly, a drop in the performance has to be noticed, when a large scale visualization is saved or loaded.

Due to the fact that all visualizations are created manually in *ProVision*, aggregated visualizations can be created to give an overview about the application landscape. Therefore, *links* (see Section 14.1.2) can be used to visualize aggregated interconnections between the different domains.

Rating: 4

14.1.7. Impact Analysis and Reporting

ProVision provides different possibilities to perform impact analyses on the data contained in the repository. These possibilities are provided by the predefined report types: *Navigation Report, Property Grid, Association Grid,* and *Navigation Grid.* Creating own report types or queries is not directly supported by *ProVision*, but can be performed by using the interface to *Crystal Reports*. Leveraging the functionalities provided by *Crystal Reports*, a variety of options for visualizations and data transformations is available to the user.

The property grid allows the user to automatically create a report containing all objects of a selected object type as well as selected properties thereof. An example of an association grid is shown in Figure 14.12.

One of the built-in capabilities for impact analysis provided by *ProVision* is the so-called *Navigation Report*. To create a navigation report, the user can select a class as base object of the report and specify which relations should be traversed to the target object. Figure 14.6 visualizes the navigation report selector, which supports the specification of the objects contained in the navigation report. The resulting report is generated automatically and cannot be edited.

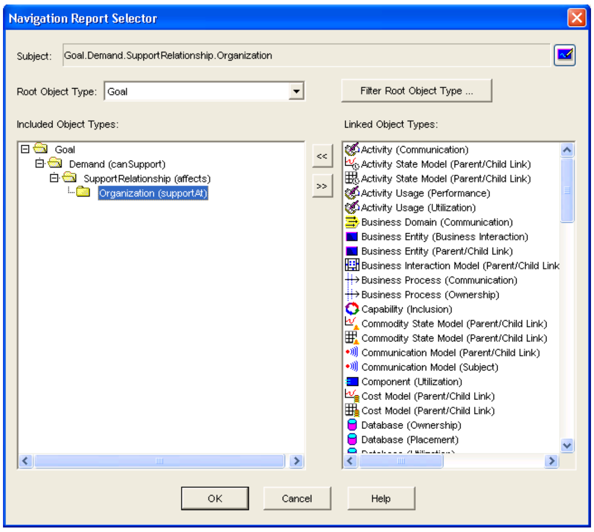


Figure 14.6.: *ProVision* - axis Impact Analysis and Reporting: Navigation report showing a root object and all existing connections

Another impact report, which is supported by *ProVision* is the so-called *Association Grid*. This grid allows the user to get an overview on the relations between two directly connected classes. Thereby, the user can delete and introduce relationships by interacting with the generated report. If an existing relationship is deleted, a warning dialog asks if the underlying data within the repository should be updated as well.

Further, *ProVision* supports a *Navigation Grid*, which allows the user to create a matrix displaying relationships between two selected classes. In contrast to the association grid, these classes do not need to have a direct relation, but can also be connected via other classes, which can be traversed. The result is visualized in a matrix, with one class as x-axis, the other class on the y-axis, and a check mark in the respective cell if a connection between the two classes exists. An example of a navigation grid is shown in Figure 14.7.

In addition to the different report types as introduced above, *ProVision* provides a neighborhood inclusion feature, which can be used within any kind of diagram. This feature allows the user to select an object visualized and automatically include all directly connected objects. Leveraging this feature, a model containing directly connected objects can easily be created.

Rating: 4

14.1.8. Usability

The first impression of *ProVision* presents an intuitive and well structured user interface. The window is divided into a main window and a menu structure provided on the left side. The menu structure contains the entries *Explorer*, *Repository*, *Inventory*, *Model Palette*, *Gallery*, and *Property*. Thereby, the explorer menu entry provides a starting point for the user giving a customizable overview about the modeling language used and the information contained within the repository.

A list of the available repositories and notebooks is provided within the *Repository* entry, which can be used to change the selected repository or notebook. Using the *Inventory* entry, the user can browse and change the elements, e. g. objects, models, or reports, contained within the underlying repository.

deployed from ApplicationSystem to DeployedApplicationSystem supportBy from DeployedApplicationSystem to SupportRelationship supportAt from SupportRelationship to Organization	Headquarter	Subsidiary Hamburg	Subsidiary London	Subsidiary Munich	Warehouse
Accounting and Costing System	✓				
Accounting System	✓				
Business Traveling System					
Business Traveling System [1.5,2.0]					
Campaign Management System	✓	✓	✓	✓	
Costing System	✓				
Customer Complaint System		✓	✓	✓	
Customer Relationship Management System	✓				
Customer Satisfaction Analysis System	✓				
Data Warehouse					
Document Management System					
Financial Planning System					
Fleet Management System					
Human Resources System					
Inventory Control System	✓	✓	✓	✓	✓
Knowledge Management System	✓				
MIS (Management Information System)	✓				
Monetary Transactions System (Germany)	✓	✓	✓	✓	✓
Monetary Transactions System (Great Britain)			✓		

Figure 14.7.: *ProVision* - axis Impact Analysis and Reporting: Navigation Grid showing the existing relationships between Application Systems and Organizational Units

Supplementary functionalities are provided within the *Model Palette* and the *Gallery* menu entries, which provides functionalities to manage the visual representations (symbols) of the objects contained within the repository on models and diagrams. Finally, the *Property* menu entry provides an overview of the properties of the currently selected object.

Maintaining data within *ProVision* can be easily done, leveraging the modeling capabilities provided. Thus, the user can easily create, update, or delete data by interacting with the graphical representation. As models are the focal point of *ProVision*, even more sophisticated modeling capabilities are provided, e. g. a strong navigation mechanism between models is supported.

Furthermore, *ProVision* comes shipped with a strong support for administrative tasks, e. g. automatic backups when closing the repository. Nevertheless, these backup mechanisms may lead to unexpected behavior, especially if an unexperienced user performs unmindful actions.

Summarizingly, *ProVision* provides a simple to handle tool, which supports a wide range of graphical models, but lacks an automation mechanism behind the creation process of these models. A novice user may need some training or advice to avoid unexpected behavior of the tool. The built-in help functionality provided by *ProVision*, which includes a guided tour and some quick start exercises as well as a glossary, is a good starting point for the prospective user.

Rating: 7

14.2. Evaluation of EA Management Support

This section describes the results of the scenario simulation for EA management support.

14.2.1. Landscape Management

Modeling different statuses of the application landscape with *ProVision* is possible, leveraging the functionality to create different notebooks. Thereby, a status, e. g. *Draft, Review, Final, Approved, Retired, Obsolete,* or *Temporary* as well as a description can be assigned to each of these notebooks. Whereas, the different notebooks contained in one repository conform to the same information model, the data and models maintained within each of these notebooks are separated. This separation allows the user on the one hand to model diverse scenarios of the evolution of the application landscape without comprising the current data. On the other hand, this approach does not provide the possibility to model relations across different notebooks, e. g. to model the relationship between an application system of the current landscape and an application system evolution in the planned landscape.

A graphical analysis of the status quo of the application landscape can easily be created leveraging the visualization functionalities provided by *ProVision*. An exemplary process support map, illustrating the current landscape, is shown in Figure 14.8. As mentioned above, future evolutions of the application landscape can be maintained and graphically modeled using an own notebook for each planning scenario, e. g. one for each planned landscape and one for the target landscape. Nevertheless, a deduction of the planned landscape from the information about projects, which drive the transformation of the application landscape, is not directly supported by *ProVision*.

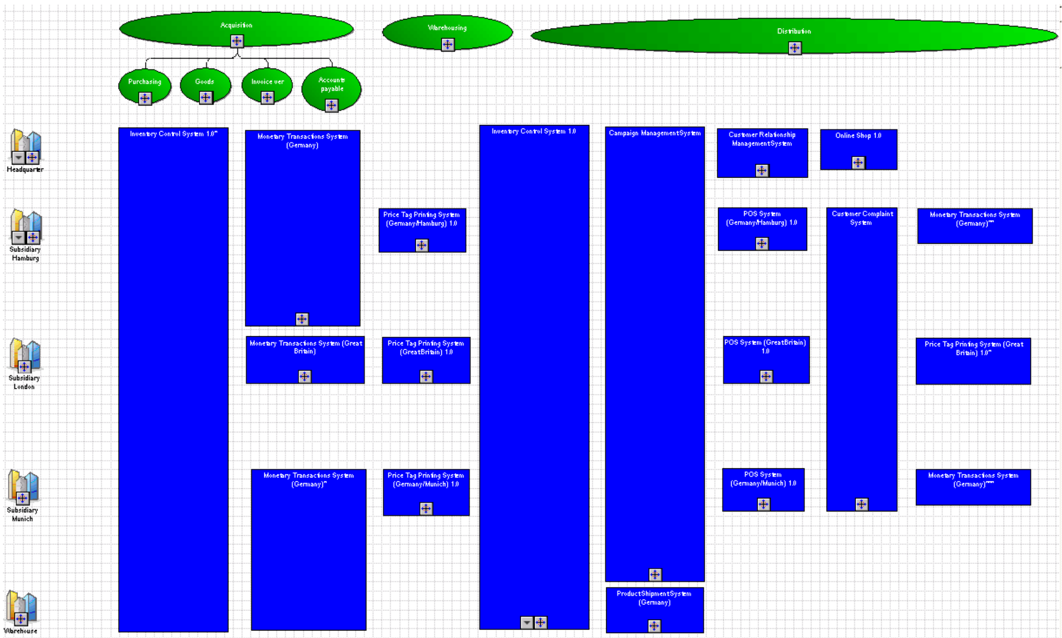


Figure 14.8.: *ProVision* - axis Landscape Management: Process support map showing the current landscape

Using different notebooks, a graphical visualization of the planned landscape (cf. Figure 14.9) and the target landscape (cf. Figure 14.10) can be created. The differences between the two variants of the application landscape can be highlighted manually using the flexible visualization techniques provided by *ProVision*. Figure 14.9 for instance visualizes the planned application landscape, highlighting the differences to the current landscape by color-coding.

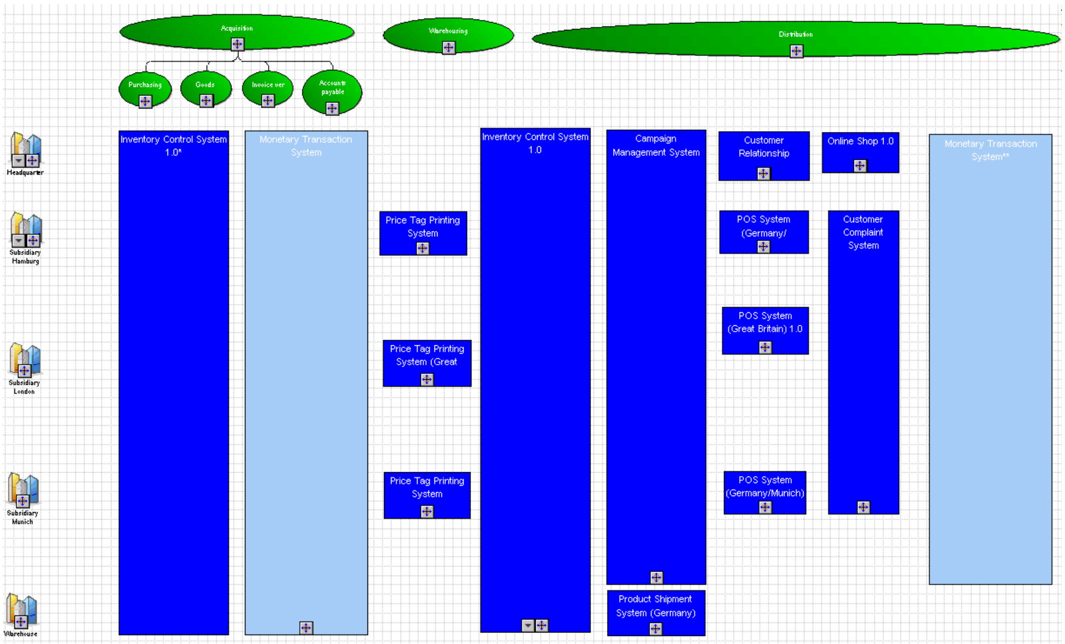


Figure 14.9.: *ProVision* - axis Landscape Management: Process support map showing the planned landscape

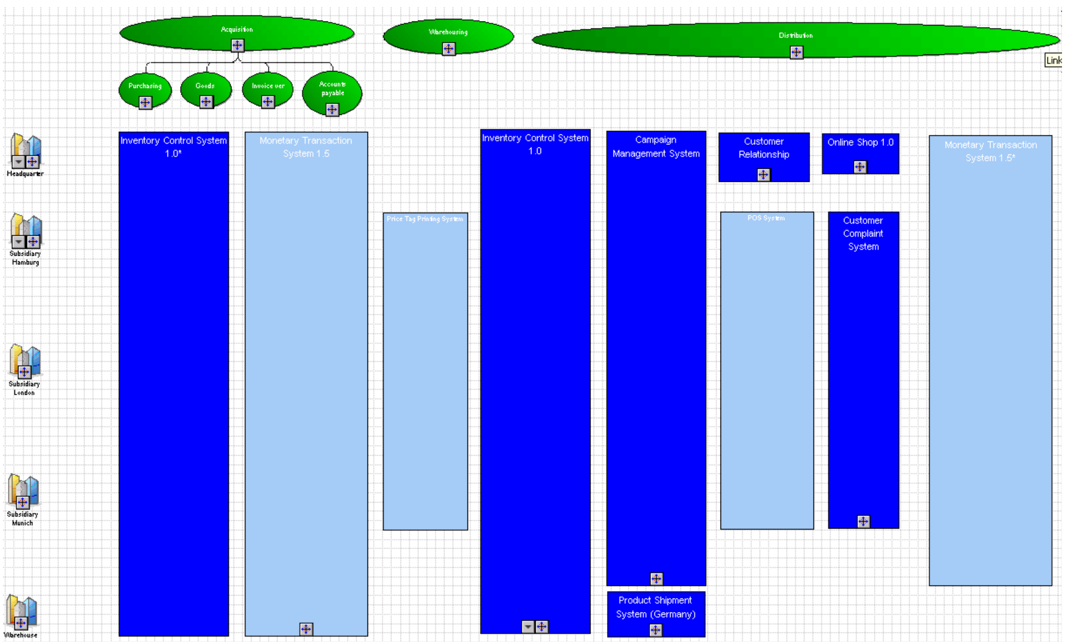
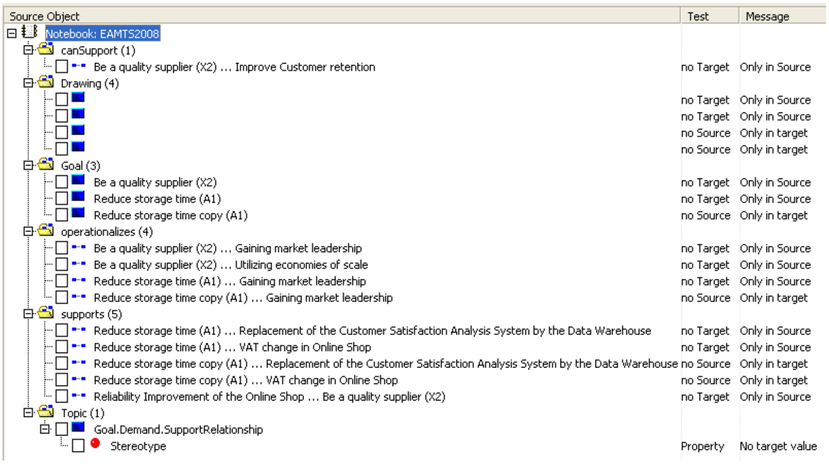


Figure 14.10.: *ProVision* - axis Landscape Management: Process support map showing the target landscape

Furthermore, *ProVision* provides a comparison functionality, which operates on two different notebooks and allows the user to list the differences of the models and/or data contained. Thereby, the user can select the elements, e. g. classes, models, or associations, which are of interest. An exemplary result of this comparison functionality is visualized in Figure 14.11.



Source Object	Test	Message
Notebook: EAMTS2008		
canSupport (1)		
Be a quality supplier (X2) ... Improve Customer retention	no Target	Only in Source
Drawing (4)		
	no Target	Only in Source
	no Target	Only in Source
	no Source	Only in target
	no Source	Only in target
Goal (3)		
Be a quality supplier (X2)	no Target	Only in Source
Reduce storage time (A1)	no Target	Only in Source
Reduce storage time copy (A1)	no Source	Only in target
operationalizes (4)		
Be a quality supplier (X2) ... Gaining market leadership	no Target	Only in Source
Be a quality supplier (X2) ... Utilizing economies of scale	no Target	Only in Source
Reduce storage time (A1) ... Gaining market leadership	no Target	Only in Source
Reduce storage time copy (A1) ... Gaining market leadership	no Source	Only in target
supports (5)		
Reduce storage time (A1) ... Replacement of the Customer Satisfaction Analysis System by the Data Warehouse	no Target	Only in Source
Reduce storage time (A1) ... VAT change in Online Shop	no Target	Only in Source
Reduce storage time copy (A1) ... Replacement of the Customer Satisfaction Analysis System by the Data Warehouse	no Source	Only in target
Reduce storage time copy (A1) ... VAT change in Online Shop	no Source	Only in target
Reliability Improvement of the Online Shop ... Be a quality supplier (X2)	no Target	Only in Source
Topic (1)		
Goal.Demand.SupportRelationship		
Stereotype	Property	No target value

Figure 14.11.: *ProVision* - axis Landscape Management: Report showing differences between two notebooks

Supporting auditing and tracking, *ProVision* provides the possibility to create snapshots of a notebook. A snapshot contains all data and models of the respective notebook and is available in read-only mode. Therefore, a snapshot can be used e. g. to historicize a status of the application landscape.

The procedure consistency is mostly given. The usage of separate notebooks allows the management of the evolution of the application landscape on a very fine grained level. Nevertheless, leveraging this functionality relationships across different notebooks cannot be maintained.

The procedure integration is mostly given. *ProVision* can make reuse of most of the data previously entered in the tool. Nevertheless, some informations already maintained by the tool, e. g. the effects caused by projects on the application landscape cannot be directly used for generating a planned landscape.

Rating: 4

14.2.2. Demand Management

Accessing received demands can be performed within *ProVision* either using the *Object Inventory* view or by creating a visual representation of the demands, e. g. via a navigation model. If, in addition to a simple list of the received demands, a report should be created containing the demands and a selection of their attributes, a property grid can be used. An exemplary report, containing the received demands and selected properties is shown in Figure 14.12.

Specifying which other elements are affected by a given demand, can be performed within *ProVision*, by creating a link between the selected demand and the respective objects, e. g. application systems, organizational units, or business processes. Thereby, the *Object Inventory* view as well as a graphical representation can be used for the specification. Creating a visualization, e. g. a graphlayout map displaying the received demands and the affected application systems, as shown in Figure 14.13, can be semi-automatically done by leveraging the neighborhood functionality as described in Section 14.1.2.

Demand	id	Qualified Name	Description	demand-urgency	expbenefits
1	40004	Adapt to government regulations	Understanding the government regulations for business applications	high	
2	40002	Homogenization	Reducing the heterogenety of usiness applications	middle	
3	40001	Improve Customer retention	The percentage of customer relationships that a business is able to maintain on a long-term basis	high-middle	
4	40005	Improve enterprise-wide knowledge management	Building an intellevtual capital strategy	low-middle	
5	40003	Reduce costs	Cost-efficient support and maintenance for business applications	high	

Figure 14.12.: *ProVision* - axis Demand Management: Report listing the received demands

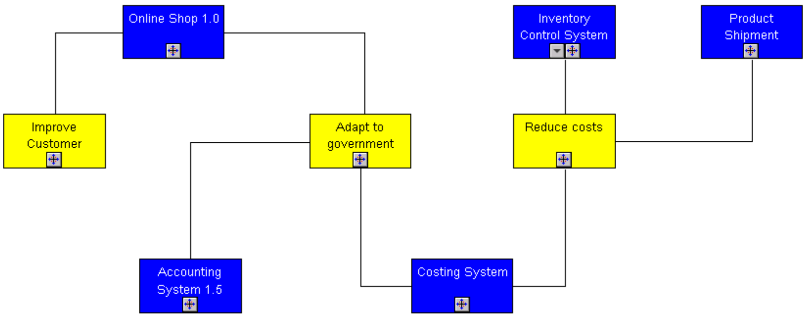


Figure 14.13.: *ProVision* - axis Demand Management: Graphlayout map showing demands and the affected application systems

The relations visualized in Figure 14.13 can be used to identify demands, which could require similar functionality and therefore be potential candidates for a consolidation into one project proposal. Due to the fact, that an automated creation of a project proposal out of one or more demands is not directly supported by *ProVision*, the user has to create a new project proposal first and then map the respective demands to the introduced project proposal.

More sophisticated reporting mechanisms, e. g. similarity reports, which list demands that affect the same application systems or require similar functionality, etc. are not directly supported by *ProVision*, but can be created using the integrated functionality to execute *Crystal Reports*.

The procedure consistency is mostly given. Some of the objectives referred to above cannot be created using the functionalities provided by *ProVision*, but can easily be created leveraging the interfaces to Microsoft Excel or *Crystal Reports*.

The procedure integration is mostly given. Although *ProVision* can reuse data previously entered into the tool, a deduction from demands to projects is not directly supported.

Rating: 4

14.2.3. Project Portfolio Management

Due to the fact, that *ProVision* focuses on modeling instead of a process guidance, no built-in process for the management of a project portfolio is provided. Nevertheless, the tool provides several ways to conduct the necessary tasks, access the required information, and achieve the needed objectives.

As alluded to above (see Section 14.2.2), displaying all received project proposals is possible either using the *Object Inventory* view or a graphical representation of the project proposals, e. g. on a *Navigator* diagram. A customizable report containing specified properties of a project can be created utilizing *Property Grids*.

Whereas, no direct support for calculating cumulative costs of a project is provided by *ProVision*, the different types of costs can be defined in the tool using the supplemental object type *Cost Elements* or customized properties. Nevertheless, utilizing the export functionalities to Microsoft Excel or *Crystal Reports* provides the possibility to calculate a variety of costing reports.

Impact analysis for deriving e.g. application systems or business processes affected by a project portfolio can be performed, and graphical visualizations as well as matrix diagrams (cf. Figure 14.14) can be created. Highlighting application systems, which are affected by selected projects, in a previously generated software map, e.g. a process support map, is not supported automatically, but can be created manually by changing the styles of the respective application systems (cf. Figure 14.9). Utilizing the integration of *Crystal Reports* portfolio matrices (cf. Figure 4.28) and conflict reports (cf. Figure 4.23) can be created, although no direct support is provided by *ProVision*.

Figure 14.14.: *ProVision* - axis Project Portfolio Management: Navigation Grid showing projects and their affected application systems

The procedure consistency is partially given. The management of project portfolios is not completely supported, especially in respect to cumulative cost calculations. Additionally, some of the objectives (Figure 4.28 and Figure 4.27) are not directly supported.

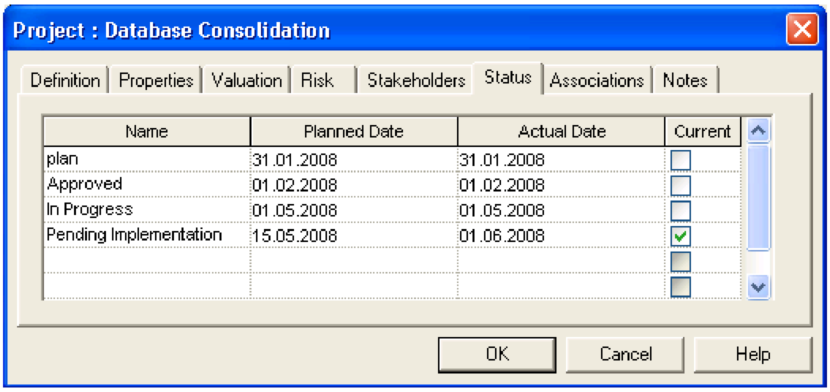
The procedure integration is mostly given. Although data about different cost types could be entered into the tool, these informations could not be reused for the conduction of cumulative cost calculations.

Rating: 3

14.2.4. Synchronization Management

ProVision provides basic support for synchronizing multiple projects currently affecting the application landscape. For every project, different statuses can be maintained within the tool, as shown in Figure 14.15. Thereby, a distinction can be made between the planned date and the actual date of a status transition. Whereas, a project delay can be simulated using the status properties of a project, a distinguished concept for project delays is not presented in the information model of *ProVision*.

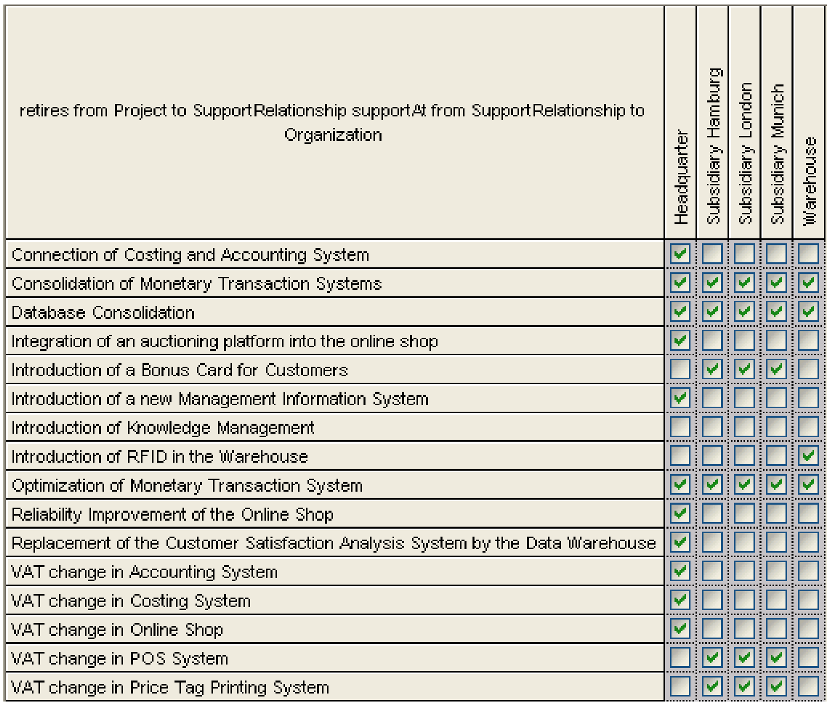
Due to the fact, that *ProVision* does not directly support project delays, an impact analysis of potentially affected other projects is not provided. Therefore, a project delay, which would be simulated in *ProVision* by changing the actual date of a project, does not cause any notifications or warnings, which point out potential candidates for projects affected by the delay. Nevertheless, *ProVision* provides the possibility to conduct impact analyses regarding affected elements, e.g. organizational units or



Name	Planned Date	Actual Date	Current
plan	31.01.2008	31.01.2008	☐
Approved	01.02.2008	01.02.2008	☐
In Progress	01.05.2008	01.05.2008	☐
Pending Implementation	15.05.2008	01.06.2008	✓
			☐
			☐

Figure 14.15.: *ProVision* - axis Synchronization Management: Screenshot visualizing the different status maintained for an project

business processes, which can be performed graphically using a graphlayout map or by leveraging the report functionality as visualized in Figure 14.16. Thereby, the report can only take one relationship, either *retires* or *introduces*[2], into consideration.



retires from Project to SupportRelationship supportAt from SupportRelationship to Organization	Headquarter	Subsidiary Hamburg	Subsidiary London	Subsidiary Munich	Warehouse
Connection of Costing and Accounting System	✓	☐	☐	☐	☐
Consolidation of Monetary Transaction Systems	✓	✓	✓	✓	✓
Database Consolidation	✓	✓	✓	✓	✓
Integration of an auctioning platform into the online shop	✓	☐	☐	☐	☐
Introduction of a Bonus Card for Customers	☐	✓	✓	✓	☐
Introduction of a new Management Information System	✓	☐	☐	☐	☐
Introduction of Knowledge Management	☐	☐	☐	☐	☐
Introduction of RFID in the Warehouse	☐	☐	☐	☐	✓
Optimization of Monetary Transaction System	✓	✓	✓	✓	✓
Reliability Improvement of the Online Shop	✓	☐	☐	☐	☐
Replacement of the Customer Satisfaction Analysis System by the Data Warehouse	✓	☐	☐	☐	☐
VAT change in Accounting System	✓	☐	☐	☐	☐
VAT change in Costing System	✓	☐	☐	☐	☐
VAT change in Online Shop	✓	☐	☐	☐	☐
VAT change in POS System	☐	✓	✓	✓	☐
VAT change in Price Tag Printing System	☐	✓	✓	✓	☐

Figure 14.16.: *ProVision* - axis Synchronization Management: Report listing projects and organizational units, which use an application system, which is affected by the respective project

More sophisticated impact analyses, e. g. identifying dependencies between projects, can be done leveraging the integration of *Crystal Reports*. This integration can also be utilized for the creation of Gantt-like diagrams, which can otherwise be manually created within *ProVision* leveraging the flexible drawing techniques provided by the tool as shown in Figure 14.17.

[2]For further information please refer to the introduction of the information model in Section 3.2.

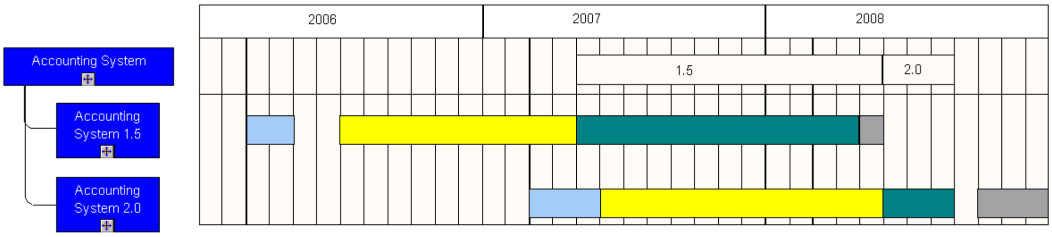


Figure 14.17.: *ProVision* - axis Synchronization Management: Time interval map visualizing the cycles of two application systems

The procedure consistency is partially given. Although *ProVision* supports the maintenance of temporal attributes, impact analyses, e. g. according to project delays and affected elements, as well as graphical analyses are not directly supported.

The procedure integration is mostly given. Some informations previously entered in the tool cannot directly be reused in this scenario, e. g. dependencies between projects cannot be identified.

Rating: 3

14.2.5. Strategies and Goals Management

Whereas, the predefined part of the information model of *ProVision* does not provide a general strategies class, but specialized classes for e. g. goals, plans, and opportunities, a custom object type can be used to introduce the general strategy class. Visualizing the goals, which operationalize a strategy, as well as the demands and affected application systems can be performed by utilizing a graphlayout map as shown in Figure 14.18. This map can be created semi-automatically using the neighborhood functionality provided by *ProVision*. Thereby, the limitations, as referred to in Section 14.1.2, must be considered.

In addition to the graphical impact analyses as alluded to above, *ProVision* supports the creation of matrices displaying correlations between strategies/goals and other objects, e. g. projects or demands. Figure 14.19 for instance displays which strategies are operationalized by which goals.

Information on the fulfillment of different goals can be stored as custom property. Performing impact analyses containing calculations regarding the achievement of goals in e. g. organizational units is not supported directly by *ProVision*, but can be easily performed leveraging the integration of *Crystal Reports* or the export functionality to Microsoft Excel.

The procedure consistency is partially given. Whereas, the tool provides good support for tracing the effects of strategies and goals, performing impact analysis especially regarding fulfillment calculations is not directly supported by *ProVision*.

The procedure integration is mostly given. Although some information previously entered into the tool, e. g. relations between different elements can be reused in this context, information about the achievement of goals cannot be aggregated on the level of organizational units.

Rating: 4

14.2.6. Business Object Management

Managing business objects and their exchange between different application systems of the application landscape is not directly supported by the built-in information model of *ProVision*, as classes as the

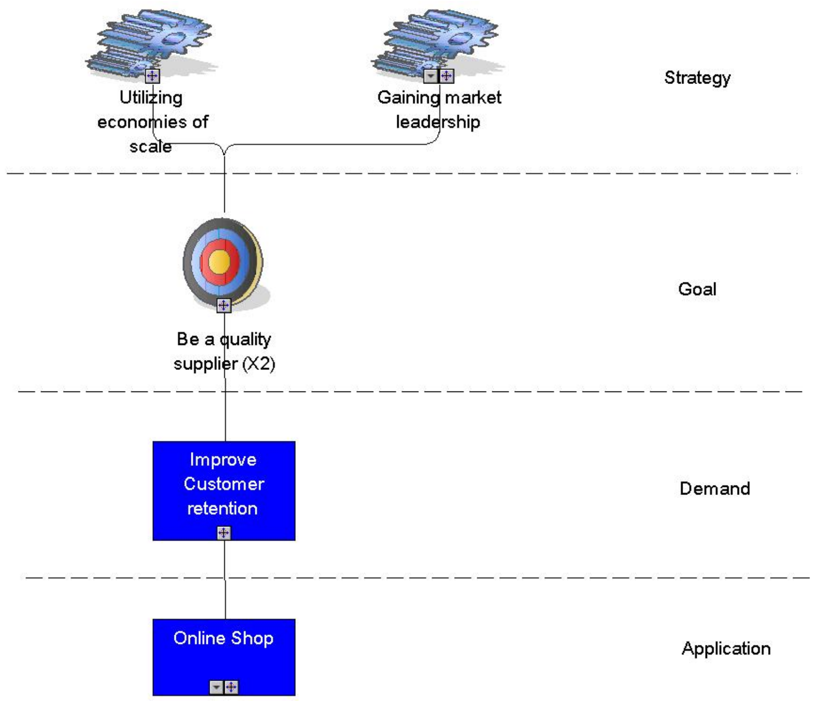


Figure 14.18.: *Pro Vision* - axis Strategies and Goals Management: Impact analysis starting with strategies

operationalizes from strategy to Goal	Be a quality supplier (X2)	Be cost effective (X1)	Increase customer satisfacti	Increase number of purchas	Reduce Complaints (D1)	Reduce storage time (A1)	Reduce time to delivery (B1)
Gaining market leadership	✓		✓	✓	✓	✓	✓
Utilizing economies of scale	✓	✓		✓		✓	✓

Figure 14.19.: *Pro Vision* - axis Strategies and Goals Management: Report listing strategies and goals

business objects are not supported. Nevertheless, such concepts can be easily introduced to the information model by leveraging the customization capabilities as introduced in Section 14.1.5. After the introduction of the new concepts, business objects, the functions performed on them as well as the interfaces and application systems performing the change can be maintained within *Pro Vision*.

Whereas, reports illustrating, if a link between two object types exists, can easily be created within *Pro Vision*, more sophisticated impact analyses are not directly supported. Figure 14.20 shows a report listing business objects and the application systems performing changes on them. More sophisticated impact analysis, e. g. regarding the exchange of business objects between application systems as well as the operations performed on them, can be created leveraging the integration of *Crystal Reports*.

manipulates from BusinessObject to Function supportWith from Function to SupportRelationship supportBy from SupportRelationship to DeployedApplicationSystem	Accounting and Costing System	Accounting System 1.5	Accounting System 2.0	Accounting System 2.5	Business Traveling System 1.0	Business Traveling System 1.5	Campaign Management System	Costing System	Customer Complaint System	Customer Relationship Management System	Customer Satisfaction Analysis System 1.0	Customer Satisfaction Analysis System 1.2	Data Warehouse	Document Management System
Customer							✓		✓	✓				
Customer Complaint									✓	✓				
Invoice														
Monetary Transaction														
Price Tag														
Stock Item							✓							
Stock Item*														

manipulates from BusinessObject to Function supportWith from Function to SupportRelationship supportBy from SupportRelationship to DeployedApplicationSystem	Financial Planning System	Fleet Management System	Human Resources System	Inventory Control System 1.0	Inventory Control System 1.0*	Inventory Control System 1.5	Knowledge Managment System	MIS (Management Information System) 1.0	MIS (Management Information System) 2.0	MIS (Management Information System) 3.0	Monetary Transaction System 1.5	Monetary Transaction System 1.5*	Monetary Transaction System*	Monetary Transactions System (Germany)
Customer														
Customer Complaint														
Invoice				✓										
Monetary Transaction														✓
Price Tag														
Stock Item				✓										
Stock Item*														

manipulates from BusinessObject to Function supportWith from Function to SupportRelationship supportBy from SupportRelationship to DeployedApplicationSystem	Monetary Transactions System (Germany)*	Monetary Transactions System (Germany)**	Monetary Transactions System (Germany)***	Monetary Transactions System (Great Britain)	Monetary Transactions System (Great Britain)*	Online Shop 1.0	Online Shop 1.5	POS System	POS System (Germany/Hamburg) 1.0	POS System (Germany/Hamburg) 1.5	POS System (Germany/Munich) 1.0	POS System (Germany/Munich) 1.5	POS System (Great Britain) 1.0	POS System (Great Britain) 1.5
Customer						✓	✓		✓	✓	✓	✓	✓	✓
Customer Complaint														
Invoice														
Monetary Transaction				✓					✓	✓	✓	✓	✓	✓
Price Tag									✓	✓	✓	✓	✓	✓
Stock Item						✓	✓		✓	✓	✓	✓	✓	✓
Stock Item*														

manipulates from BusinessObject to Function supportWith from Function to SupportRelationship supportBy from SupportRelationship to DeployedApplicationSystem	Price Tag Printing System	Price Tag Printing System (Germany/Hamburg) 1.0	Price Tag Printing System (Germany/Hamburg) 1.5	Price Tag Printing System (Germany/Munich) 1.0	Price Tag Printing System (Germany/Munich) 1.5	Price Tag Printing System (Great Britain) 1.0	Price Tag Printing System (Great Britain) 1.0*	Price Tag Printing System (Great Britain) 1.5	Product Shipment System (Germany)	Supplier Relationship Management System	Worktime Management System (Germany/Hambur	Worktime Management System (Germany/Munich)	Worktime Management System (Great Britain)
Customer													
Customer Complaint													
Invoice													
Monetary Transaction									✓				
Price Tag													
Stock Item									✓				
Stock Item*													

Figure 14.20.: *ProVision* - axis Business Object Management: Report listing business objects and the application systems performing changes on them

In addition to the reports as referred to above, graphical representations of business objects and information flows between application systems, as shown in Figure 14.21, can be created, leveraging the flexible visualization techniques provided by *ProVision*.

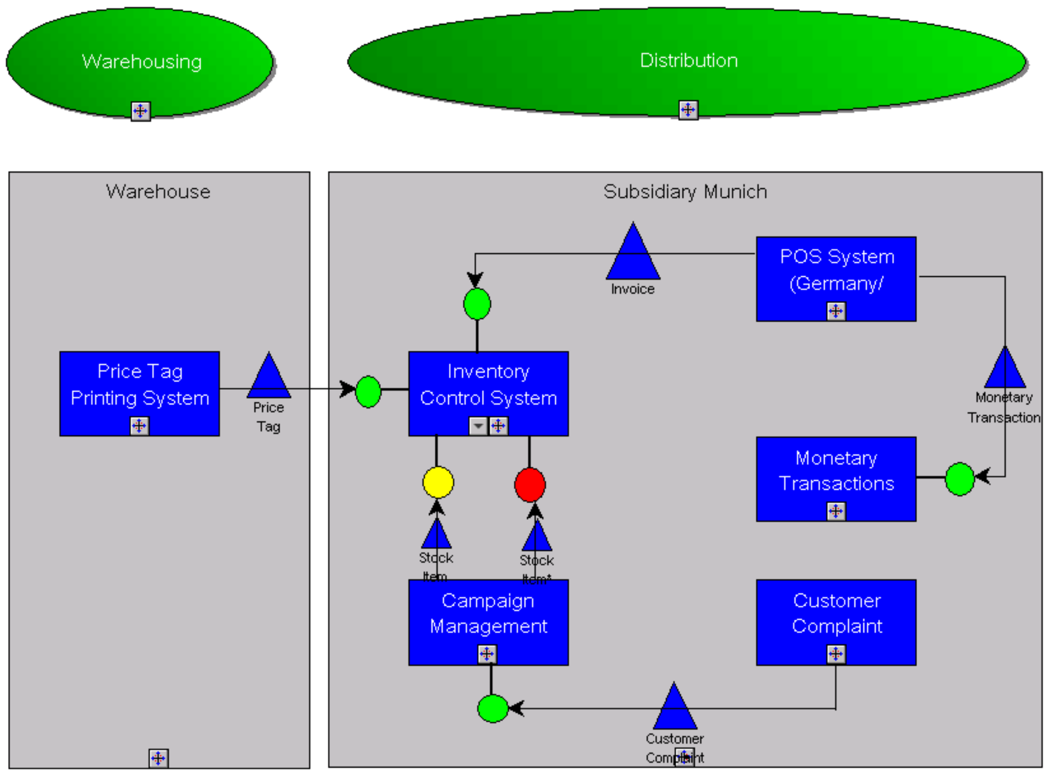


Figure 14.21.: *ProVision* - axis Business Object Management: Cluster map illustrating the business objects, which are exchanged between application systems

Further information, e. g. concerning which application system holds the master copy of a business object, can be easily maintained within the tool, via an extension of the underlying information model. A similar customization is necessary to support storing the information about manual operations.

The procedure consistency is mostly given. The missing concepts can be easily introduced using custom objects. Although, manual effort was required, most of the objectives can be achieved.

The procedure integration is completely given. Information previously entered in the tool can be reused in this context.

Rating: 4

14.2.7. SOA Transformation

The information model of *ProVision* provides a predefined concept for maintaining services. Thereby, a minor customization is necessary regarding the support of specifying service level agreements (SLAs) for services. Following the hybrid approach for a SOA transformation as described in Section 4.2.7, further minor adaptations to the information model have to be performed, e. g. a custom property has to be defined in order to store information about the type (standardized vs. differentiating) of business processes or the frequency of changes of an application system. After customization of the information model most of the objectives of the scenario can be created using the functionalities provided by *ProVision*.

A process support map can be created manually, which highlights the business processes according to their type. Using this visualization an impact analysis can be conducted analyzing the business processes and their supporting application systems. This visualization can be used to annotate additional information, e. g. the frequency of changes of an application system by color-coding as visualized in Figure 14.22. Due to the fact, that *ProVision* does not directly support a modification of the background-color of a symbol according to the values of a property[3], the annotation has to be performed manually by the user.

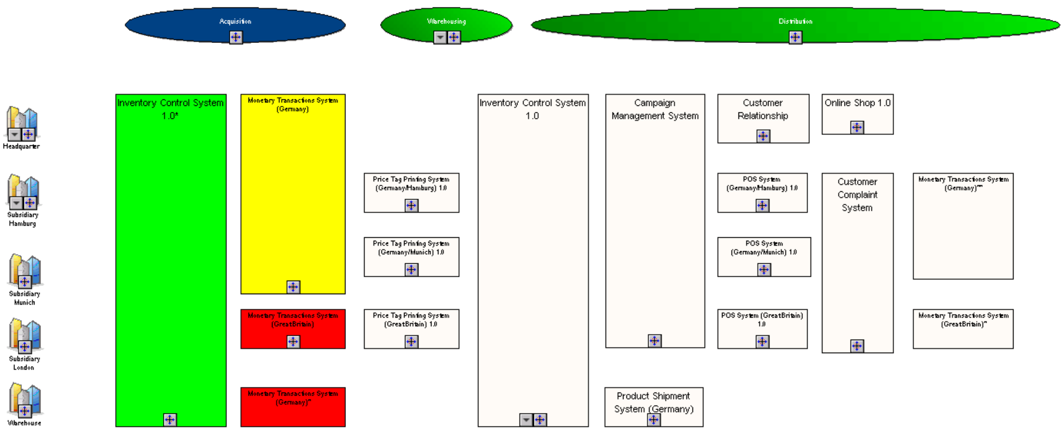


Figure 14.22.: *ProVision* - axis SOA Transformation: Business process map showing differentiating business processes and supporting application systems

Conducting an impact analysis to identify application systems, which will be affected by changes in the near future, is easily possible, leveraging the visualization capabilities provided by *ProVision*. Using the neighborhood inclusion functionality, application systems and the respective projects, which will perform changes on the application systems, can be displayed. Furthermore, demands could be included to extend the considered time scope. Reports containing the number of usages of an application system within different domains, or which operations are performed on an business object by a

[3]See Section 4.1.4 for more details on annotating visualizations.

specified application system, are not directly supported by *ProVision*, but can be created leveraging the export functionalities to Microsoft Excel or the integration of *Crystal Reports*.

ProVision supports different possibilities to provide information about the impacts the SOA transformation will have on the current application landscape. As alluded to in Section 14.2.1, *ProVision* provides the possibility to model the evolution of the application landscape in a distinct notebook, a copy of the current one, and conduct user-defined comparisons on two notebooks. If a more fine-granular view on the changes is needed, time interval maps as the one shown in Figure 14.23 can be used. Although, time interval maps displaying temporal information on the application systems and the migration to services are not directly supported by *ProVision*, such visualizations can be created manually using the flexible visualization techniques provided.

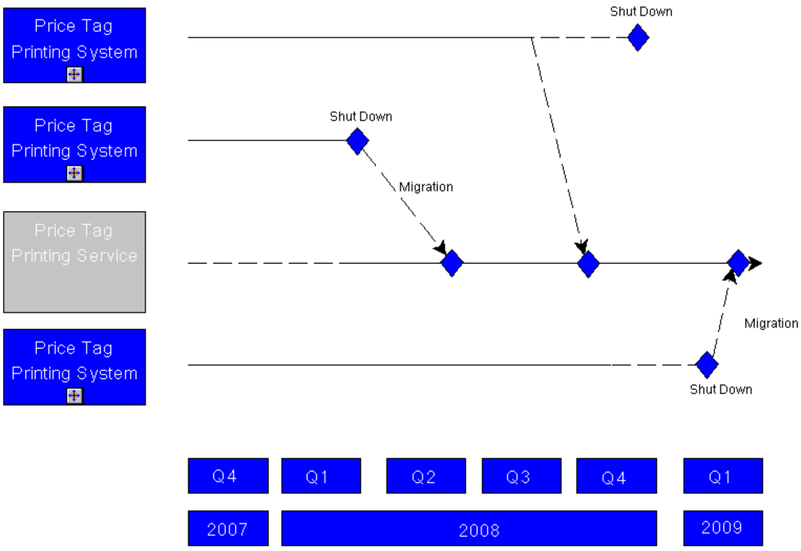


Figure 14.23.: *ProVision* - axis SOA Transformation: Time interval map showing the migration of functionalities from an application system to business services

The procedure consistency is partially given. The simulation of the scenario requires minor adaptations on the information model.

The procedure integration is completely given. Information entered in preceding scenarios can be reused in this context.

Rating: 4

14.2.8. IT Architecture Management

ProVision does not come shipped with a built-in support for IT architecture management. Therefore, the concepts of architectural blueprints, architectural solutions, blueprint elements, and solution elements are not supported in the predefined information model. Utilizing the customization functionalities provided by *ProVision*, the concepts alluded to above and the respective relationships can be introduced.

Accomplishing impact analysis regarding the conformance to architectural solutions can be performed, using different possibilities: A graphical analysis of the conformance can be done manually using a cluster map as visualized in Figure 14.24. Whereas, simple matrices, e. g. listing architectural solutions and their usage within different organizational units can be created using the functionalities provided by *ProVision* (cf. Figure 14.25). Reports covering several information, e. g. architectural solutions, solution elements, and the application systems using them, are not directly supported but can be created leveraging the export functionalities to Microsoft Excel or the integration to *Crystal Reports*. Visualizing the occurrence of the different architectural solutions is not directly supported by *ProVision*, although by utilizing the export functionalities to Microsoft Excel bar charts illustrating this context can be created.

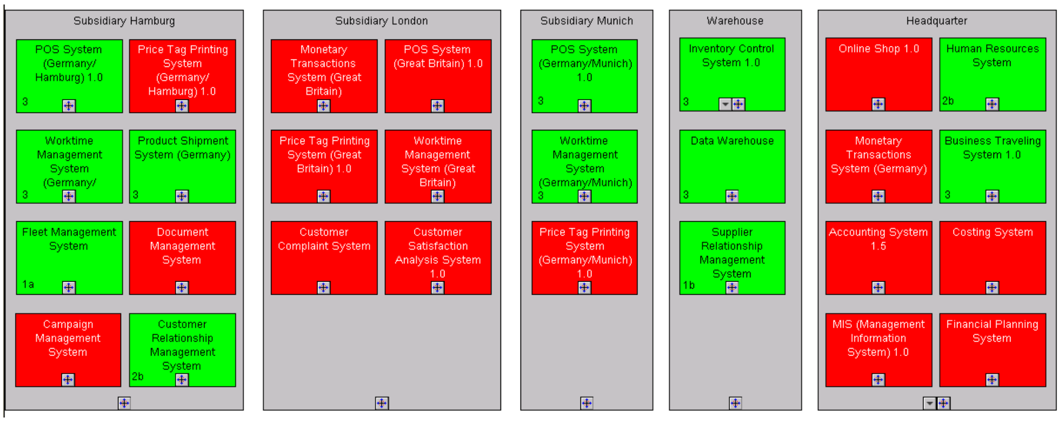
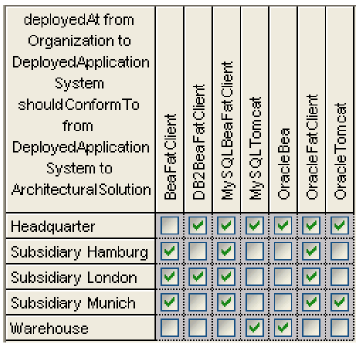


Figure 14.24.: *ProVision* - axis IT Architecture Management: Cluster map showing application systems and their conformance with architectural solutions



deployedAt from Organization to DeployedApplicationSystem shouldConformTo from DeployedApplicationSystem to ArchitecturalSolution	BeaFatClient	DB2BeaFatClient	MySQLBeaFatClient	MySQLTomcat	OracleBea	OracleFatClient	OracleTomcat
Headquarter	☐	☑	☑	☑	☑	☑	☑
Subsidiary Hamburg	☑	☐	☑	☐	☐	☑	☐
Subsidiary London	☑	☑	☑	☐	☐	☑	☐
Subsidiary Munich	☑	☐	☑	☐	☐	☑	☑
Warehouse	☐	☐	☐	☑	☑	☐	☐

Figure 14.25.: *ProVision* - axis IT Architecture Management: Navigation Grid showing which architectural solution is used in which organizational unit

The procedure consistency is partially given. Although, the concepts used in this scenario are mostly not supported by the predefined information model of *ProVision*, they can be introduced and most of the deliverables can be created.

The procedure integration is completely given. Information previously entered, especially from landscape management, can be reused in the context of this scenario.

Rating: 4

14.2.9. Infrastructure Management

ProVision provides built-in support for managing different infrastructure elements, e. g. database management systems and technologies, but does not provide a general *infrastructure service* class.

Although, the scenario as described in Section 4.2.9 focuses on database management systems, similar analyses could be performed regarding other infrastructure elements. For the simulation of the scenario, the classes as introduced in Section 14.2.8 are used.

Getting an overview about the database systems currently used can be performed by manually filtering the solution elements according to their relationship to the infrastructure service *database*. A report listing the database management systems and selected attributes thereof, as shown in Figure 14.26, can be created using the association grid functionality provided by *ProVision*.

Solution Element	Qualified Name	endIntroduction	endPhaseOut	endProduction	id	startIntroduction	startPhaseOut	startProduction	supportEndDate
1	DB2 6.0	31/05/2005	31/12/2010	31/12/2009	240008	01/06/2004	01/01/2010	01/06/2005	30/06/2011
2	MySQL 2.1	30.09.2005	01.06.2010	15.06.2009	240001	01.01.2005	16.06.2009	01.10.2005	31.12.2010
3	Oracle 9i	31/12/2005	31/12/2014	31/12/2013	240005	01/04/2005	01/01/2014	01/01/2006	31/01/2015

Figure 14.26.: *ProVision* - axis IT Architecture Management: Report listing the databases and selected aspects

An impact analysis listing databases, which are running out of support is not directly supported by *ProVision*, but can be conducted manually viewing the information stored for each database, as listed in Figure 14.26. Visualizing the used databases for the application systems can be done leveraging the reporting functionalities or creating a graphical representation manually (cf. Figure 14.27).

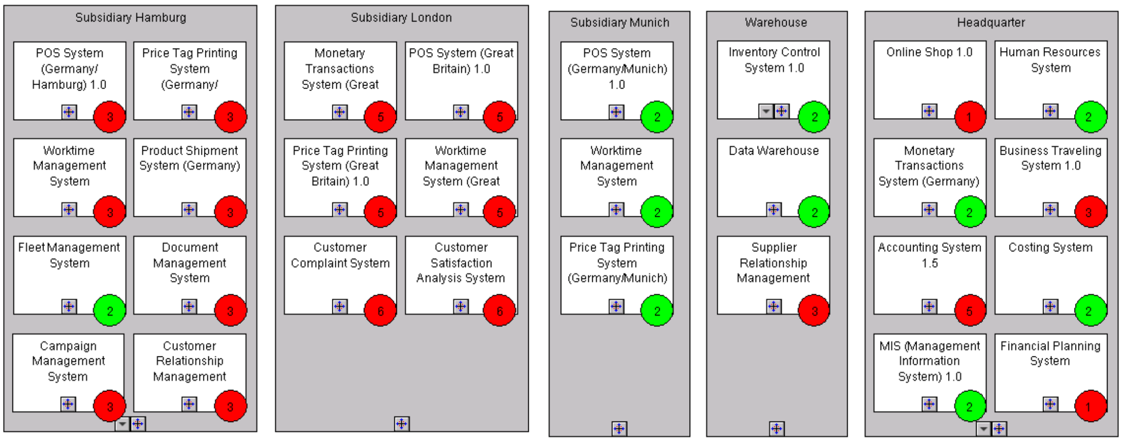


Figure 14.27.: *ProVision* - axis IT Architecture Management: Cluster map visualizing databases running out of support and the application systems using them

Additional information e. g. operating, maintenance, or licensing costs can be stored for each infrastructure service using customized attributes. Whereas, these different costs can be stored and maintained within *ProVision*, a report containing calculated costs is not directly supported, but can be created leveraging the export functionalities to Microsoft Excel or *Crystal Reports*.

Although analyses regarding the lifecycle information of infrastructure elements are not directly supported by *ProVision*, different capabilities to perform impact analyses regarding relationships between distinct elements of the application landscape are provided. Leveraging these functionalities, a report can be created illustrating the application systems, which relies on a database management system that should be replaced. Furthermore, a graphical representation, e. g. a cluster map can be used to visualize the impact of these replacement on the organizational units, that host the affected application systems, as shown in Figure 14.28.

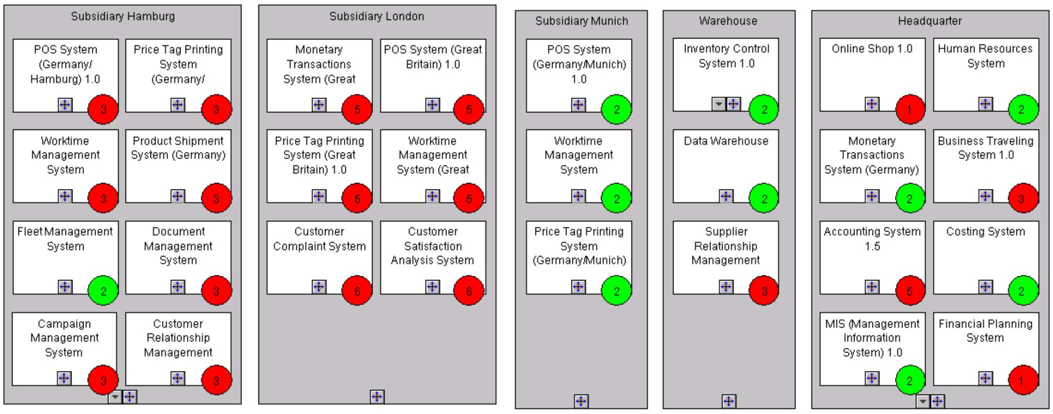


Figure 14.28.: *ProVision* - axis IT Architecture Management: Cluster map visualizing application systems and databases affected by consolidation

The procedure consistency is mostly given. Whereas, some information, e. g. lifecycles of database management systems can be stored and maintained in *ProVision*, performing impact analysis on these information is not directly supported.

The procedure integration is completely given. The data previously entered into the tool can be reused in this context.

Rating: 3

14.3. Tool Vendor's Profile

Ulrich Meyer, Account Executive - Channel Sales Central Europe

About Metastorm

With a focus on enterprise visibility, optimization, and agility, Metastorm offers market-leading solutions for Enterprise Architecture (EA), Business Process Analysis & Modeling (BPA) and Business Process Management (BPM). As an integrated product portfolio, Metastorm Enterprise™ allows organizations to maximize business results by unifying strategy, analysis and execution. Metastorm is the only solution provider to bring together these critical disciplines on a single software platform to enable an understanding of enterprise architecture and strategy, accurate impact and opportunity assessment, effective process execution, and accelerated value realization for organizations worldwide.

Metastorm ProVisionEA provides a complete suite of modeling tools for both enterprise architects and business analysts. Key enterprise assets - including systems, data, resources, products, and suppliers - and their inter-dependencies can be modeled, shared, and refined in a standalone or collaborative environment. Metastorm ProVisionBPA provides robust business process analysis and enterprise modeling capabilities to define critical business processes and associated dependencies, facilitate requirements analysis, simulate multiple scenarios, and optimize processes and related enterprise assets against strategic objectives using Six Sigma, SCOR, ITIL, or other optimization frameworks. Metastorm BPM® is a highly scalable, enterprise BPM suite designed to support automation, deployment, integration, analysis, monitoring, and improvement of both human and system-based processes within and across organizations.

Each solution in the Metastorm Enterprise portfolio can operate standalone or in combination, and Metastorm's Common Interchange Format (CIF) and support for industry standards ensures interoperability with other technologies. Combined these applications provide unmatched enterprise visibility and agility - allowing business and IT users of all levels to accurately assess the impact of key decisions, identify new opportunities for improvement, and optimize overall performance for strategic advantage.

For more information visit www.metastorm.com.

CHAPTER 15

Telelogic AB (Telelogic System Architect)

Contents

Telelogic provides the EA management tool *System Architect*, leveraging a flexible approach for EA modeling, emphasizing both the IT and business side. The following sections describe a detailed analysis of *System Architect* Version 11.0, on the one hand taking the perspective of different functionalities provided by the tool and on the other hand evaluating the support provided by *System Architect* concerning different tasks within the EA management process.

15.1. Evaluation of Specific Functionality

15.1.1. Importing, Editing, and Validating

Importing data into *System Architect* can be performed from different import formats, e. g. CSV, XML, or XMI. Sources like Microsoft Excel are supported indirectly by using CSV-formats. Thereby, the structure of the import file can easily be derived from an empty export file of the corresponding definition. Data transformation capabilities, e. g. splitting comma separated values, are not directly provided by *System Architect* but can be leveraged using the integrated Microsoft Visual Basic for Applications (VBA) capabilities or Microsoft Excel.

The concepts defined within the information model of *SoCaStore* (see Section 3.2) can all be supported by *System Architect*, except for the floating point numbers, which are mapped to strings accordingly. The imported data is validated during the import process according to aspects as e. g. primary keys, data types, or data length. At the end of the import process, a log-file summarizing the import results is provided as shown in Figure 15.1. Different kinds of conflict resolution strategies are provided to avoid collisions during importing. Examples for these strategies are: Never replace existing definition (default), delete all fields then add new data, update single fields when data supplied, update single fields - clear field, if no data is supplied in the file to be imported.

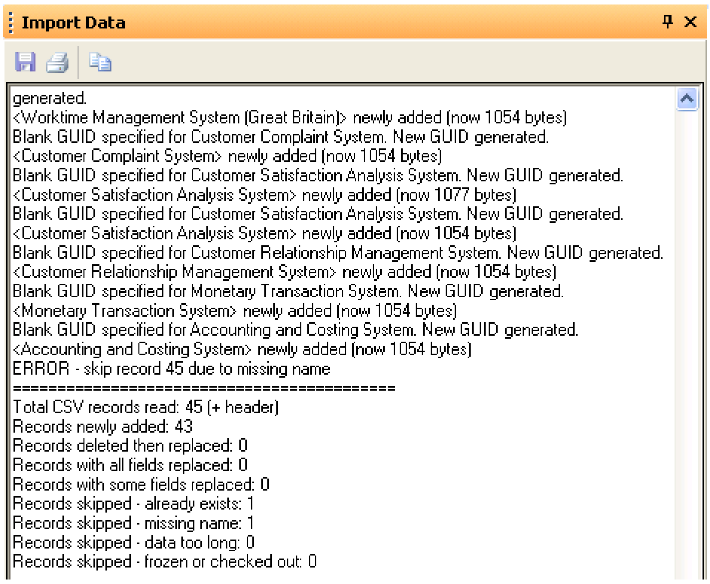


Figure 15.1.: *System Architect* - axis Importing, Editing, and Validating: Log-file summarizing the import results

The *System Architect* provides different possibilities to edit the data contained in the repository. Data can be edited directly by selecting the respective element in the navigator or picking it by double-clicking in a visualization. Besides, more sophisticated editing mechanisms are provided, e. g. matrix browsers can be used to view, analyze, and edit relationship data. Thereby, the matrices can be defined directly by the customer using a predefined dialog. An example of a matrix browser visualizing goals (as columns), strategies (as rows) and existing operationalize relationships (as crosses in the respective cells) is shown in Figure 15.2. If the matrix editor is used to maintain relationships, the displayed relationship information in the dialogs of the respective elements should be set to read-only to avoid data inconsistencies.

Strategy / Business Goal	Be a quality supplier (X2)	Be cost effective (X1)	Increase customer satisfaction (D2)	Increase number of purchases (C1)	Reduce Complaints (D1)	Reduce storage time (A1)	Reduce time to delivery (B1)
Gaining market leadership	X	X	X	X	X	X	X
Utilizing economies of scale	X	X		X		X	X

Figure 15.2.: *System Architect* - axis Importing, Editing, and Validating: Matrix browser showing relationships between strategies and goals

Further detailing the documentation of elements, each object contained in the repository can be linked with external documentation (e. g. Microsoft Word documents, pictures, or diagrams). Additionally, using the XMI interface provided by *System Architect*, UML models can be imported or exported. Leveraging the functionality provided by the integrated VBA scripting engine, various ways to import or export data to and from *System Architect* can be used to access every object contained in the repository.

Rating: 5

15.1.2. Creating Visualizations

System Architect provides a set of predefined diagram types, e. g. UML diagram types, business process diagrams, or organization charts. Additionally, the possibility to specify user-defined diagrams is provided. Thereby, the name of the diagram, the objects, which can be visualized in, and the symbols representing an object, can be defined. Leveraging this customization capabilities, the GUI of *System Architect* is adapted, according to the selected diagram type, allowing the user to add only symbols defined for the distinct diagram type. If, for instance, a cluster map is selected, the user can only create symbols referring to application systems, organizational units, or locations. In addition, some symbols for adding text, rectangles, pictures, and comments are provided for each diagram type.

The different diagrams can be created and elements can be displayed on the canvas by either leveraging the drag and drop functionality with the respective definitions contained within the explorer, by using the choices dialog from the context menu or by selecting the corresponding button and latter clicking on the canvas. Thereby, the respective symbol is created and a dialog is opened, which allows the user to either select an existing element from the repository or create a new one. The style of a symbol, e. g. font, text position, or line as well as the graphic itself, can be changed on each diagram without affecting other occurrences of the symbol.

System Architect provides different functionalities to support the creation of a visualization. Concerning the manual positioning of symbols, grids, or rulers are provided to support the alignment of objects. In addition, automated layout possibilities, e. g. network, hierarchically, or in a circle, are provided. The layout of lines can be modified accordingly utilizing bend points or predefined routing algorithms, e. g. center-to-center routing.

A cluster map (see Figure 15.3), process support map (see Figure 15.12 and Figure 15.14), or swim lane diagram (see Figure 15.24 and Figure 15.23) can be implemented by defining the respective diagrams and the contained object types in the customization file. Latter, the visualizations can be created manually utilizing the drag and drop functionality. Thereby, the relative positioning of elements does not have a semantic meaning[1], e. g. the nesting of an application system within the symbol representing an organizational unit in Figure 15.3 was achieved manually using the flexible drawing techniques provided by *System Architect*.

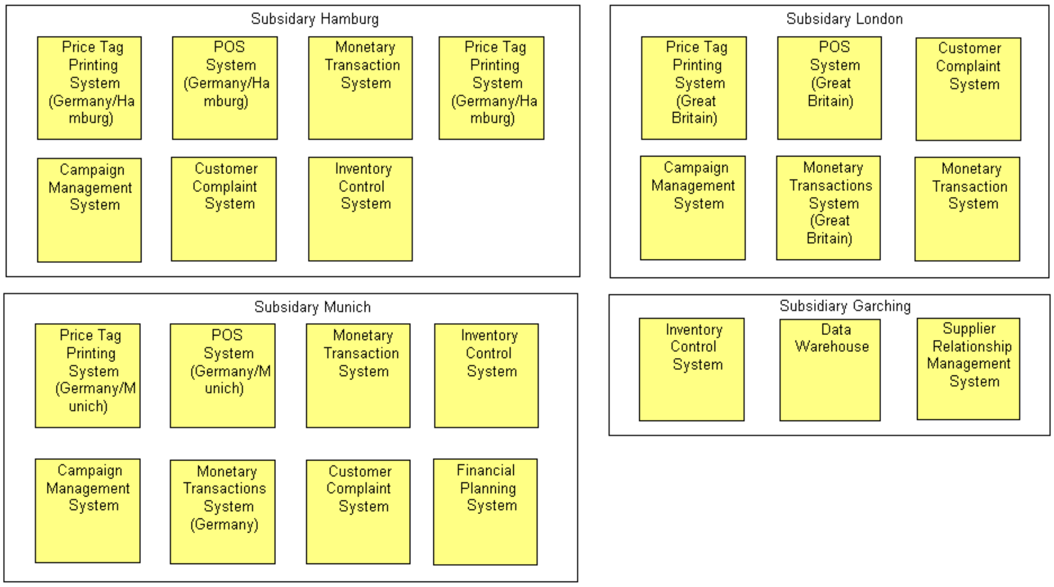


Figure 15.3.: *System Architect* - axis Creating Visualizations: Cluster map

Similarly, a graphlayout map can be created. Although *System Architect* does not directly support the automated generation of lines between application systems to represent interconnections, these lines can be created manually and linked to the respective data contained in the repository. Figure 15.4 shows an exemplary graphlayout map visualizing interconnections of the application system *Online Shop*.

Time interval maps and portfolio matrices are not directly supported by *System Architect*[2]. Nevertheless, these visualization types can be created manually or semi-automatic via VBA using e. g. pictures as backgrounds, which can be utilized to show the planning interval (see Section 15.2.4 for an example of time interval maps). Although portfolio matrices cannot be created automatically by *System Architect*, the functionality to use grids as the background of visualizations can be utilized to create a portfolio matrix manually or semi-automatic via VBA (see Section 15.2.3 for examples).

Furthermore, *System Architect* supports a diagram type called *explorer diagram*. This diagram type is similar to our graphlayout map, as it visualizes elements and *immediate relatives* thereof[3]. An ex-

[1] There are some diagram types provided by *System Architect*, where the positioning of an element holds a semantic meaning, e. g. swim lane diagrams.

[2] *Telelogic* provides an own product for product and project portfolio management called *Focal Point*, which supports among other functionalities the creation of time interval maps and portfolio matrices.

[3] In addition to the possibility to visualize immediate relatives of a selected symbol, different reports can be utilized to e. g. display indirect relatives over several levels of relationships. For a detailed description of

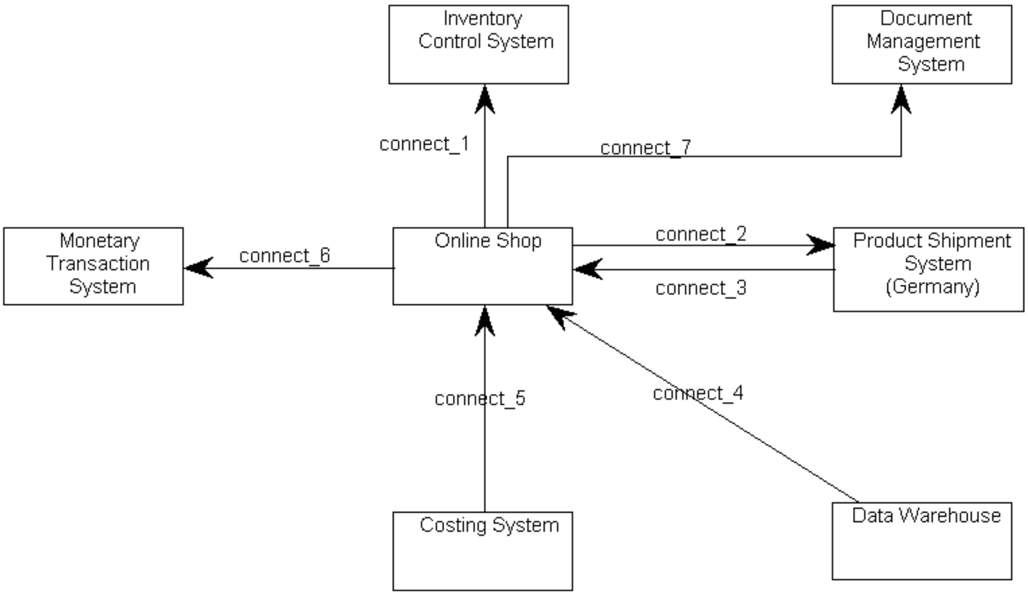


Figure 15.4.: *System Architect* - axis Creating Visualizations: Graphlayout map

emplary visualization is given in Figure 15.5. This visualization can be created semi-automatically, by selecting one or more visualized elements as starting points and latter utilizing the built-in functionality of *System Architect* to include directly connected elements and the respective links automatically.

Matrix diagrams can additionally be created automatically leveraging the built-in functionalities of *System Architect* (cf. Figure 15.2). These diagrams can be used to visualize relations between classes of the information model, e. g. strategy and goals. Thereby, these classes present the x- and y-axis of the corresponding matrix and the respective cells display an x, if the corresponding goal operationalizes the respective strategy. Furthermore, the intersection cell can represent a class, which carries additional information and attributes that can be displayed in the cell.

Summarizingly, *System Architect* provides flexible graphical functionalities. Nevertheless, the automated creation of many visualization types is not supported as out-of-the-box functionality, but can be achieved by leveraging the VBA scripting functionalities or by buying an add-on as e. g. the *SOA solution* (see Section 15.1.5). Utilizing these functionalities requires some training, but can be managed by a skilled user. Thereby, the documentation provided by *System Architect*, especially concerning the different classes, are helpful.

Rating: 5

15.1.3. Interacting with, Editing of, and Annotating Visualizations

Visualizations created with *System Architect* can be edited in various ways. The symbols displayed in a visualization can be changed concerning e. g. their style, their position, their size, or the overall layout, as detailed in Section 15.1.2. Altering the underlying information contained in the repository can be performed by selecting the respective symbol on a diagram and editing the information directly, e. g. renaming the element or utilizing the context dialog, if data not visualized should be modified. Due to the fact that the relative positioning on most diagrams does not carry semantics, changes performed

these reporting functionalities please refer to Section 15.1.6.

on visualizations regarding the positioning of symbols do not affect the data contained in the repository. Likewise, changes concerning the visualization, e. g. deleting or redirecting a link representing an interconnection does not change the underlying information, which may lead to inconsistencies between the information displayed in visualizations and the underlying repository. Nevertheless, a VBA macro can be defined, which is triggered on diagram save and writes the changed information in the repository. Visualizations can be automatically updated according to the data contained in the repository. Thereby, symbols not referring to an object of the repository are not affected by the update.

Furthermore, *System Architect* provides different possibilities to interact with visualizations, e. g. zooming or mouseover highlighting of the *Symbol, Symbol + Connected,* or *Symbol + Connected + Next* as shown in Figure 15.5. Besides, the automated graphical annotation of visualization with additional information representing attribute values is partially supported by *System Architect*, utilizing the concept of *analytic depictions*. These analytic depictions can be used to e. g. add a red database symbol to visualize a database running out of support (cf. Section 15.2.9 for an example). Thereby, the analytic depictions are based on additional attributes, which are configured for a symbol diagram

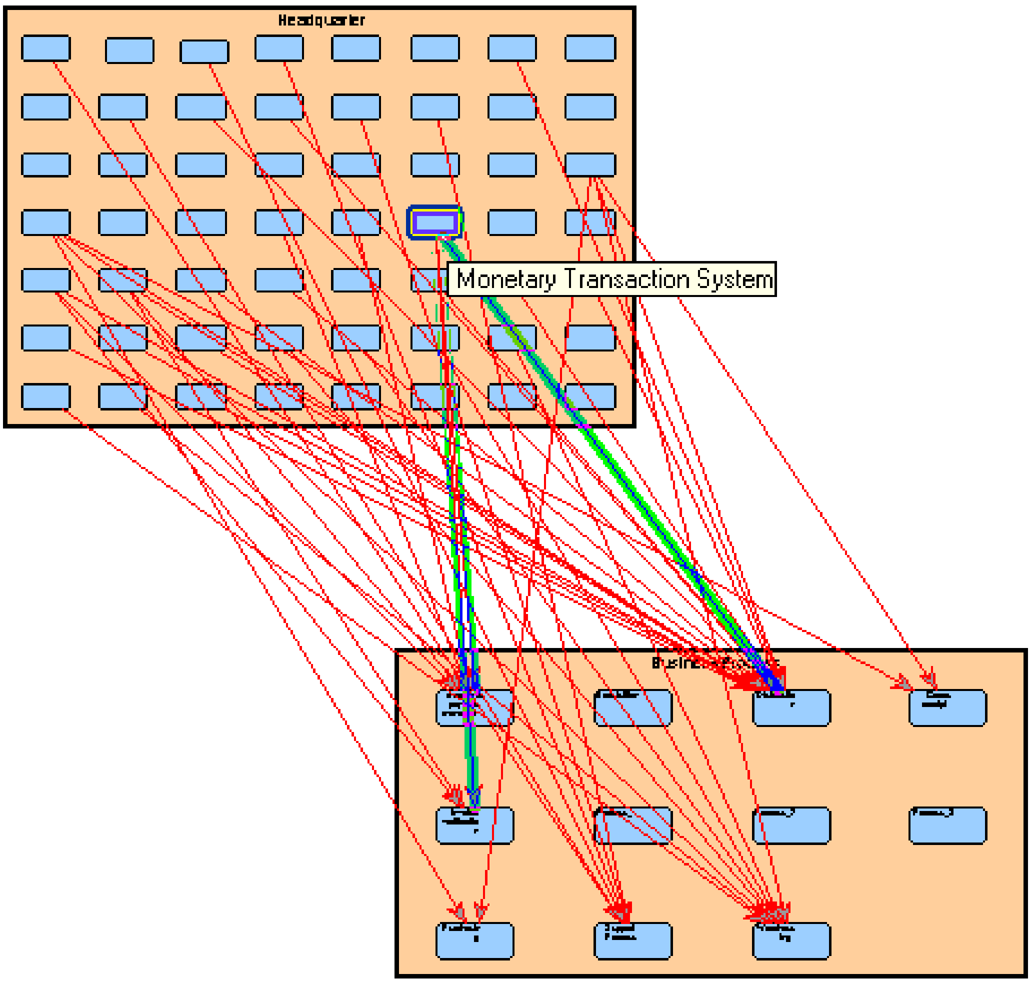


Figure 15.5.: *System Architect* - axis Interacting with, Editing of, and Annotating Visualizations: Explorer diagram highlighting directly connected elements of a selected one

combination in the definition of the information model. Setting the respective attribute value of the corresponding symbol has to be performed manually during the creation of the visualization or by using the integrated scripting engine to automatically set the additional attributes according to the attribute values of the respective elements. The thereby created visualizations offer the possibility to show or hide the depicted analytic symbols.

Changing the background color according to attribute values is not directly supported by *System Architect*, visualizations like the one shown in Figure 15.6, which illustrate the maintenance costs by color-coding, can be created manually using the flexible visualization techniques. Nevertheless, a skilled user could leverage the functionalities provided by VBA scripting to generate annotated visualizations according to attribute values.

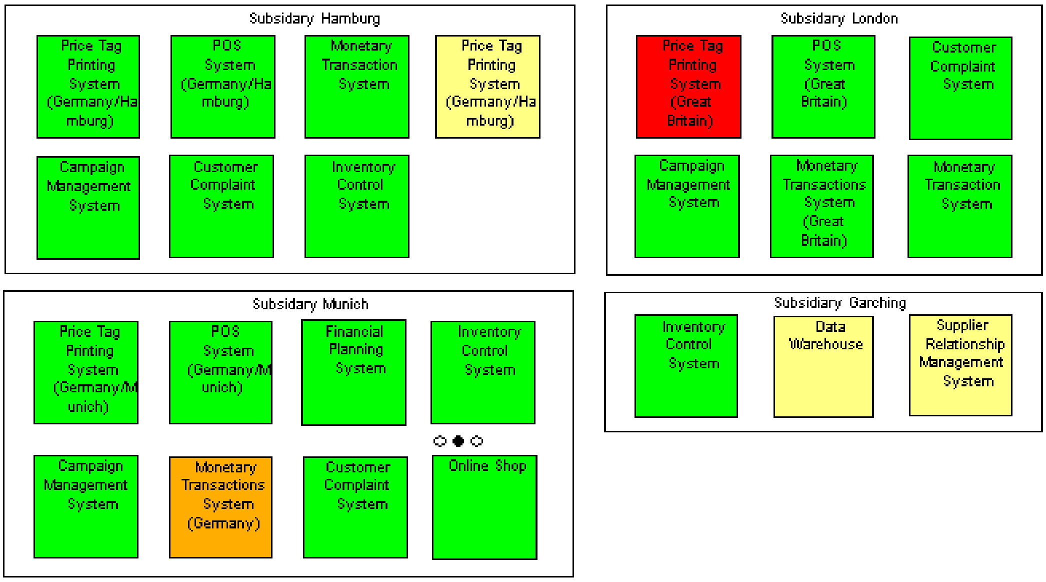


Figure 15.6.: *System Architect* - axis Interacting with, Editing of, and Annotating Visualizations: Cluster map visualizing maintenance costs by color-coding

Although filtering or layering to reduce the number of visualized symbols on a diagram is not supported by *System Architect*, except in the case analytic depictions are used, an object displayed can contain a built-in link to a so-called *child diagram*. Thereby, a mechanism to reduce the complexity of a visualization and to allow quick navigation from one visualization to another is provided. In Figure 15.6 *Online Shop* has an attached child diagram, which details its interconnections to other application systems.

Rating: 4

15.1.4. Communication and Collaboration Support

To improve communication and increase the awareness of the EA management endeavor, *System Architect* provides different possibilities to allow web-based access to information and diagrams stored within the repository. The easiest way to allow accessing information over the web is to leverage the reporting functionality to create static HTML pages. The report wizard supports the selection of diagrams, elements, and attributes that will be included in the export. Thereby, the diagrams can be exported as JPG, PNG, or SVG files. The generated HTML files can be adapted according to corporate design using templates and cascading style sheets.

The second possibility is to use a paid Add-on of *Telelogic*, the so called *System Architect XT*. This web-client supports role-based access to the information stored in the repository. The standard welcome page gives an overview about the user's last modifications, lists the tasks defined for the current user as well as some predefined reports. Furthermore, the repository and the diagrams can be browsed directly or utilizing the search engine provided. Additionally, bookmarks as well as links can be created and the complete user interface can be customized by using CSS and XSLT.

Finally, a third possibility to support the web-based access to information is provided using the so called *SA Publisher*[4]. The *SA Publisher* allows to create reports and browse the information using customizable navigation.

To support multiple users working collaboratively, *System Architect* allows via the *Catalog Manager* the specification of *roles* and *users & groups* to facilitate role-based access control to functionalities, menus, encyclopedias, and object types. For authenticating users either the *windows integrated security* can be used or *user name and password* for a *Microsoft SQL Server* or *Oracle database*. In addition, different permission levels are supported, e. g. *read, write, create,* and *destroy*, as visualized in Figure 15.7. An inheritance structure can be used to specify permissions.

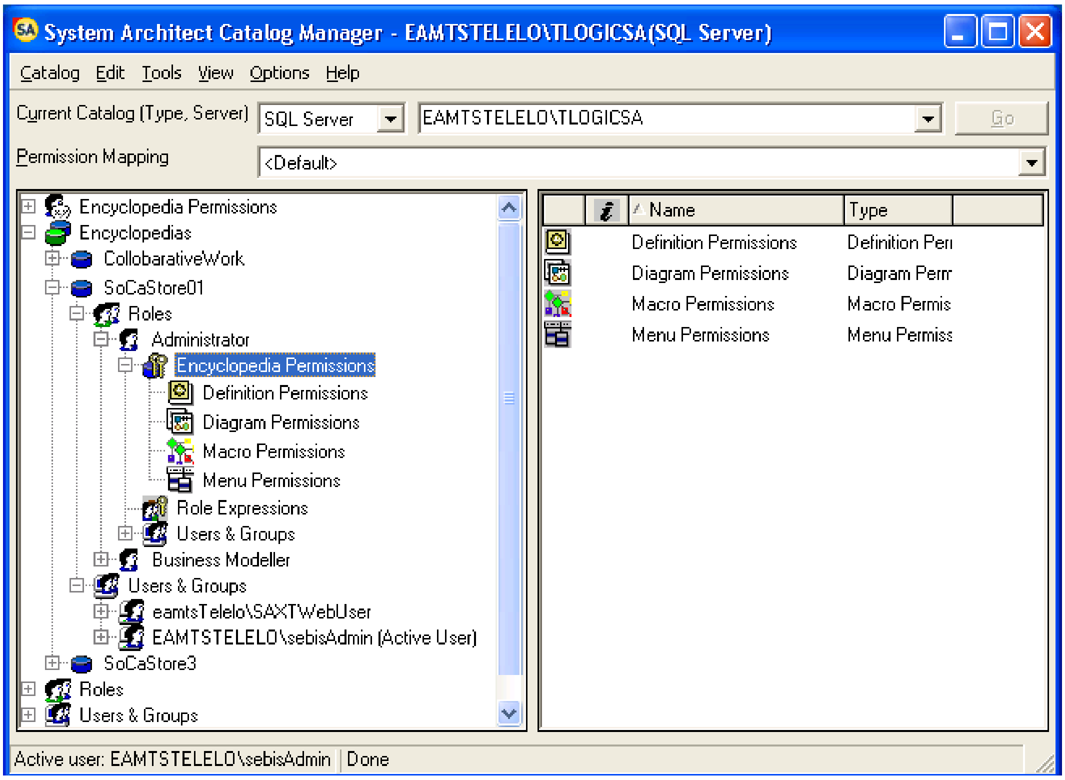


Figure 15.7.: *System Architect* - axis Communication and Collaboration Support: Screenshot of the *System Architect Catalog Manager*

Whereas the specification of organization specific workflows and their management is not directly supported by *System Architect*[5], *task items* can be defined for a user containing information, e. g. description, priority, status, start, and due date. Using the *System Architect SX* these task items

[4]The *SA Publisher* was not evaluated in detail during the survey.

[5]According to Telelogic there will be an integration of *Telelogic Change* in the upcoming release of *System Architect*. Leveraging this integration, customizable workflows can be specified and automated notifications created.

defined for the current users will be shown in a dialog. In the desktop solutions they can be accessed and maintained over the explorer.

To avoid conflicts, if multiple users simultaneously work on the data, *System Architect* provides locking functionalities. The user can check-out data from the encyclopedia, thus the respective data is locked and latter released, if a check-in is performed. Thereby, the locking is performed on item level (objects and/or visualizations) and is exclusive. A permanent locking can be achieved by using the so called *freeze function*. Besides the locking functionality provided by *System Architect*, multiple users can work together using different repositories and utilizing the provided *merge* functionality, which supports the user during the resolution of potential conflicts, if two or more repositories are brought together.

Rating: 4

15.1.5. Flexibility of the Information Model

Supporting a very flexible approach for customization, the *System Architect* provides strong functionalities concerning the adaptability of the information model using a declarative scripting language native to *System Architect*. Thereby, the entities contained within the repository of *System Architect* are classified into three types: *Definition types* representing the different entity types as e. g. application systems, business process, or organizational units, the *diagram types* defining visualizations as e. g. cluster maps, process support maps, or graphlayout diagrams, and finally the *symbol types* specifying the graphical representation of a definition type on a selected diagram type as e. g. chevron, rectangle, or line.

Providing support for customization, *System Architect* offers the possibility to select an information model out of a set of predefined information models (e. g. TOGAF, Zachman) or to start with an own model. Thereby, new classes can be introduced to the information model by renaming one of the predefined customization user classes. A multitude of typed attributes (e. g. boolean, date, numeric, text) as well as the declaration of attributes as mandatory is supported, while lacking the possibility to use floating point numbers. Nevertheless, functionalities are provided to use and define enumerations.

The *System Architect* provides the possibility to introduce new relationships either utilizing attributes that are references to another entity within the repository or by creating an own definition type for the relationship. Using the latter possibility, the relationship can be enhanced with attributes. If the first possibility is used, the relationship may refer to more than one definition type and supports the following multiplicities: 0..1, 1, 0..*, 1..*. Furthermore, *System Architect* provides functionalities for the user to define the property dialogs according to customer needs. Additionally, it can be defined by the user, which entities and relationships, if defined as definition type, can be displayed on which diagram using which symbol.

A visual representation of the information model or selected parts of it can be automatically generated as visualized in Figure 15.8, which allows the user to adapt existing elements and relationships graphically. Finally, it has to be noted that the configuration of the information model of *System Architect* is split into two files: One containing the predefined parts of the information model defined by *Telelogic*, the other including the customizations of the information model made by the user. In order to reduce complexity, the customization file can additionally be separated into single files. Leveraging this strict separation, an upgrade to a newer version of *System Architect* can be easily performed by importing the information model customization file to the new version.

In order to reduce the manual effort, *Telelogic* provides the possibility to buy add-ons as e. g. the *SOA solution*, which contain standard customizations. Utilizing this add-ons, predefined information model concepts as well as predefined reports are provided. The *SOA solution* for instance, contains predefined concepts as services, components, domains, and their attributes as well as predefined reports and automated visualizations, illustrating domains and the used services as cluster map. In addition,

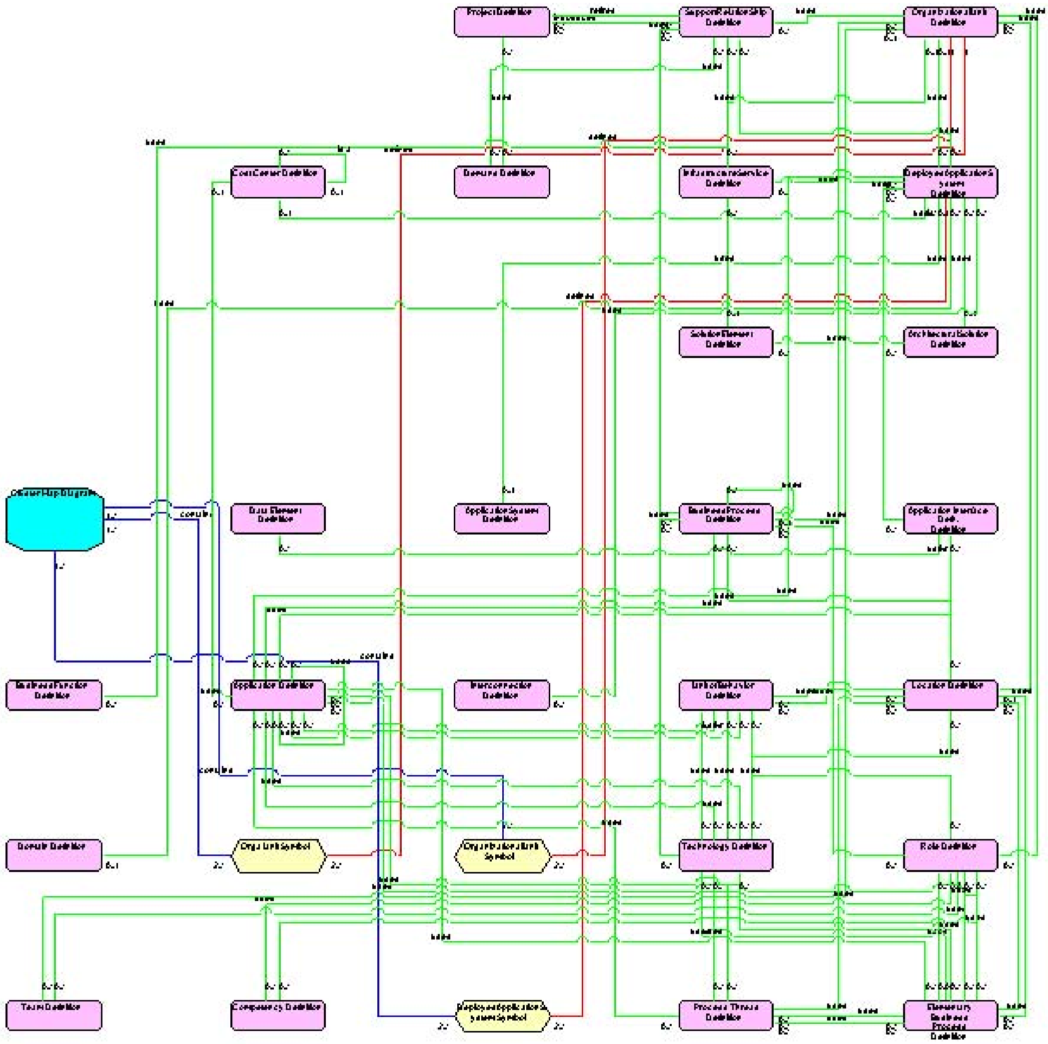


Figure 15.8.: *System Architect* - axis Flexibility of the Information Model: Graphical representation of an extract of the information model

an *analytic query builder* is provided, which supports a wizard-based creation of impact analyses. Further solutions via paid add-ons are available for e. g. NAF, DODAF, FEAF, eTOM.

Rating: 6

15.1.6. Support of large scale Data

Large scale data sets concerning thousands of application systems and their interconnections can be imported into *System Architect*, e. g. from Microsoft Excel. Although no live performance can be expected, the import of data is done in a reasonable time and the tool provides information on the progress already achieved during the import. The handling of large scale data sets is not visibly affected. Elements can be created, read, updated, or deleted in real time. Performing modifications on more than one element concurrently can be done by leveraging the export functionality to Microsoft Excel, changing the data there, and latter importing the modified data again.

Creating visualizations of the application landscape containing a large amount of data can be performed within *System Architect* considerably fast. However, during the interaction with the visualizations containing a large amount of data a drawback in the performance must be noted.

For the creation of a cluster map containing all application systems of the large scale data set two different ways must be considered: On the one hand, the built-in explorer diagram can be used for the creation. Thereby, the object type (here the *DeployedApplicationSystem*) has to be dragged and dropped on the canvas, which will result in an automated visualization of all objects contained. The result can be enhanced by a user-defined report, leveraging the *report generator*, which visualizes interconnections of the application systems. On the other hand, the scripting capabilities of VBA can be used to define an aggregated visualization (in respect of application systems and the interconnections thereof) of a cluster map, which contains links to *child diagrams* explicating the aggregated element. Such report definitions call for a skilled user in order to be created.

Besides the visualizations of the application reports can be created containing all application systems and selected aspects thereof. The execution of such reports is not visibly affected by the amount of data contained.

Rating: 5

15.1.7. Impact Analysis and Reporting

System Architect provides different ways to perform impact analyses and reporting on the data contained in the repository. In addition to the predefined reports shipped with the tool, the user can create customized reports. A simple to use way to define reports is provided by the HTML reporting functionality. Thus, static HTML reports containing selected elements and diagrams can be created[6]. Another possibility is to use the *Microsoft Word Reports* feature, which provides predefined reports concerning e. g. auditing, diagrams, objects. Thereby, the Microsoft Word reports are based on the classes of the predefined information model and require programming effort to support the customized classes.

Customized reports can be created using the *report generator* (see Figure 15.9), which provides a graphical user interface to define SQL-like queries. Thereby, classes from the predefined information model as well as customized classes can be queried. The results of the report can either be a text or an HTML file. Although transitive impact analyses can be performed using the *report generator* due to the fact that joins are supported, the possibilities using the graphical user interface are limited to joins of up to three consecutive element types. Nevertheless, joining more element types can be performed leveraging the possibility to edit the query in a textual way. Whereas selections according to element types and properties as well as ordering of elements by attribute values is supported, the capabilities concerning aggregation functions is limited to count and sum.

More sophisticated report functionalities, e. g. queries performing calculations based on elements or their attribute values can be performed using the integrated VBA scripting engine or leveraging the functionalities of Microsoft Excel via the OLE DB access to the repository. Utilizing VBA every element contained in the repository can be accessed, including customized elements. Nevertheless, creating queries using VBA requires a certain programming effort from the user. In order to spare the everyday user from employing such sophisticated functionality, *System Architect* supports the creation of reports separately from their (repeated) execution. Thereby, the results will be displayed in e. g. Microsoft Excel or Microsoft Word.

Besides the different reporting functionalities as alluded to above, *System Architect* provides a number of possibilities to perform impact analyses on visualizations. Thereby, especially the explorer diagram is of interest. This diagram provides the possibility to automatically *show immediate relatives* of a

[6] For a more detailed description of the HTML reporting functionality see Section 15.1.4.

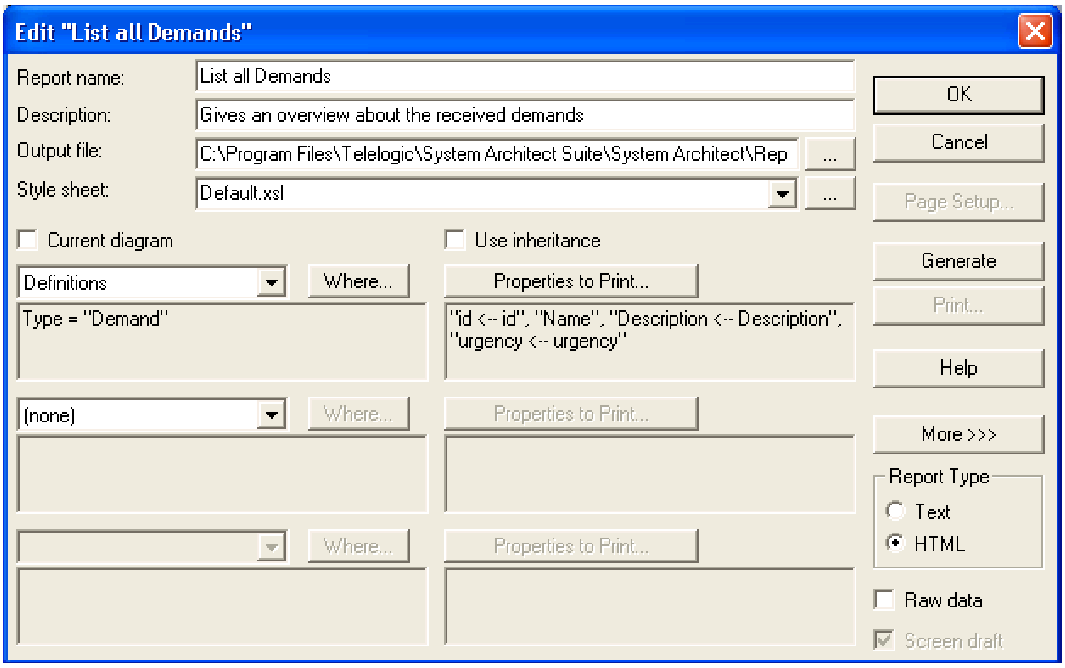


Figure 15.9.: *System Architect* - axis Impact Analysis and Reporting: Creating a report using the *report generator*

selected symbol. Utilizing this functionality, a very intuitive way to perform graphical impact analyses is given. Further graphical reporting functionalities on *explorer diagrams* are provided using so called *explorer object reports* or *explorer relationship reports*. Whereas the *explorer object report* can be used to populate an explorer diagram, e. g. with all application systems that are hosted at a certain organizational unit, the *explorer relationship report* can be used to support traversing over more than one relationship to a selected object. In addition, utilizing the mouse-over functionality of *System Architect* relationships of a selected symbol can be highlighted.

Rating: 6

15.1.8. Usability

System Architect provides a well structured and intuitive user interface. In this user interface all elements contained in the repository are displayed and organized in a tree-like structure in the *explorer view*. To simplify the access to information and to prevent unskilled users from the complexity of displaying all elements contained within the repository, different views can be created, which describe subareas of EA management, e. g. business layer, application layer, and infrastructure layer, and display only the respective elements. These views and the respective subareas can be customized according to the requirements of an organization utilizing the *view explorer*, which supports the categorization of diagrams and definitions within package-like structures.

Although *System Architect* does not provide the built-in possibility to change data by interacting with a visual representation thereof, as mentioned in Section 15.1.3, the data in the repository can be modified by double clicking the respective symbol in a diagram, selecting it from the explorer view, or utilizing the VBA capabilities to support graphical editing. The context dialogs can be configured utilizing the information model adaptation capabilities as described in Section 15.1.5. The editing of data, e. g. maintaining the *operationalize* relationship between a strategy and a business goal can be performed using a pick-list of available choices and the provided drag and drop functionality of *System Architect*. This functionality can also be used to ease the creation of visualizations. Maintaining relationships between different elements can additionally be performed using a matrix browser as described in Section 15.1.1.

Furthermore, *System Architect* supports the methodology and visualizations of pre- or user-defined EA frameworks, like TOGAF or Zachmann, to improve usability. Thereby, a visualization can be stored within *System Architect*, which offers the possibility to specify which elements or visualizations should be edited according to the selected part of the methodology (c.f. Figure 15.10. In addition to the documentations explaining the adaptability of the information model and the extendability utilizing VBA, *System Architect* comes shipped with *guidebooks* (c.f. Figure 15.11) containing manuals and tutorials, which provide a good starting point to understand the functionalities provided by *System Architect*. Thereby, these *guidebooks* can be customized by the organization to offer additional ease of use for occasional users.

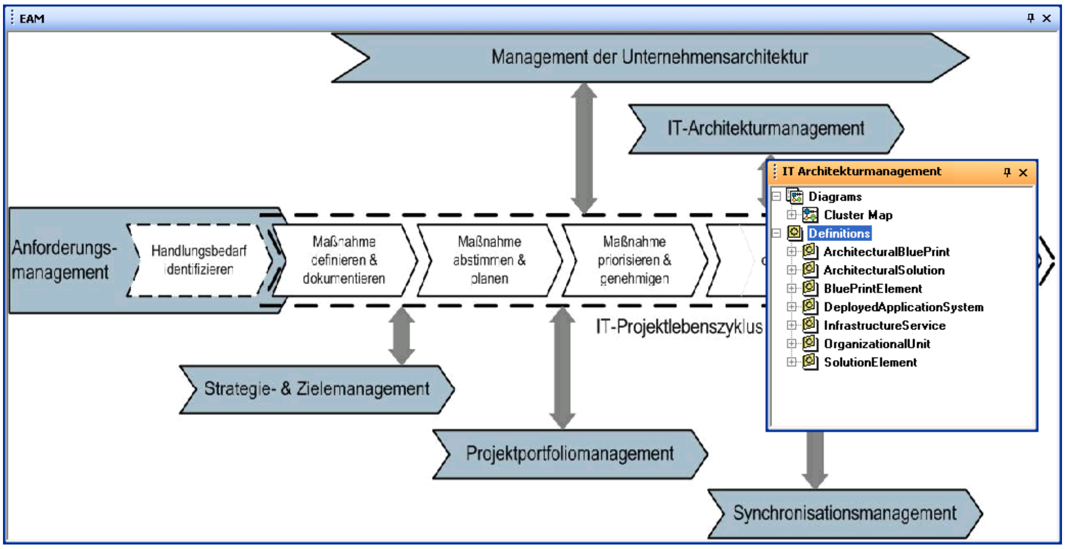


Figure 15.10.: *System Architect* - axis Usability: Screenshot of a customized methodology *System Architect*

Taking all this into account, *System Architect* provides an intuitive and easy to use repository with a user interface supporting a multitude of graphical models. Maintaining information and creating visualizations can be performed easily, although manual effort is required. If an automation in the creation process of visualizations should be achieved, a certain programming effort by a skilled user is necessary. Similar considerations hold for the definition of customized reports, which can be specified by an untrained user after a short time of familiarization.

Rating: 5

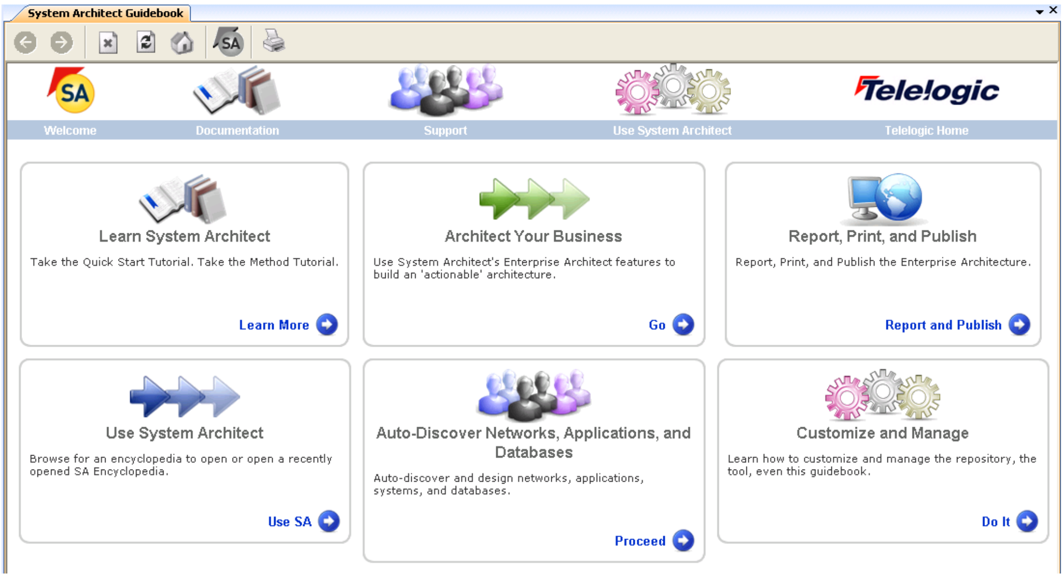


Figure 15.11.: *System Architect* - axis Usability: Screenshot of the welcome screen of the *Guidebook* shipped with *System Architect*

15.2. Evaluation of EA Management Support

This section describes the results of the scenario simulation for EA management support.

15.2.1. Landscape Management

System Architect provides different possibilities to model the evolution of the application landscape including the management of different statuses (*current, planned,* and *target*) as well as variants thereof[7]. The first possibility is to use separated encyclopedias to model different statuses of the application landscape or variations. Thus, on the one hand avoiding naming conflicts caused by e. g. an organizational unit, which has the same name in the current and planned landscape but are constructed differently. On the other hand, lacking support in the creation of reports or impact analysis across encyclopedias[8].

As the scenario specially focuses on the possibility to compare the different statuses of the application landscape, the second possibility to maintain all versions of the application landscape in one encyclopedia and store temporal information for every object (object types and diagrams) to derive the affiliation to a certain status is used. Thereby, different variants of the application landscape can be created and maintained graphically utilizing the flexible visualization capabilities provided by *System Architect*.

The predefined information model supports most of the concepts needed to accomplish the management of the application landscape including the possibility to store temporal information for every object. A concept similar to the *SupportRelationship*, which models the support provided by a provider, e. g.

[7]According to Telelogic, the next release will include a so called *Partitioning* concept, which provides the possibility to substructure an encyclopedia. These subareas could then be used to model different statuses of the application landscape. Thereby, each subarea will provide an own namespace to prevent naming conflicts between different subareas.

[8]Inter encyclopedia reports or impact analysis can be performed utilizing the tool *SA Compare*, which was not evaluated in this survey.

an application system or a service, for a business process and the usage of this support at a specific organizational unit, is not covered by the predefined information model but can easily be introduced leveraging the customization capabilities provided by *System Architect* (see Section 15.1.5).

A graphical visualization of the current landscape (see Figure 15.12) can be created utilizing the information contained in the repository. Similar visualizations illustrating different status of the application landscape, e. g. see Figure 15.13 for the planned and Figure 15.14 for the target landscape can be created and maintained within *System Architect*. Thereby, the positioning and highlighting of elements on the visualizations has to be performed manually or a paid add-on has to be utilized, which supports the generation of certain visualizations, like e. g. a process support map.

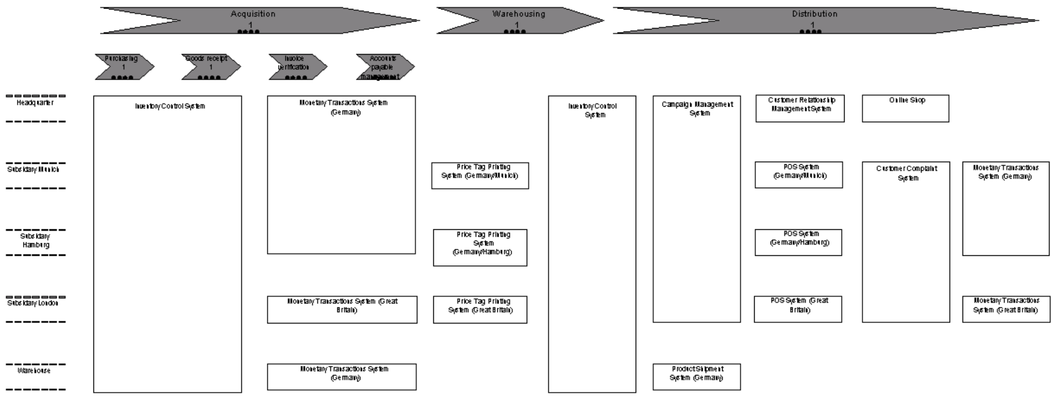


Figure 15.12.: *System Architect* - axis Landscape Management: Current landscape

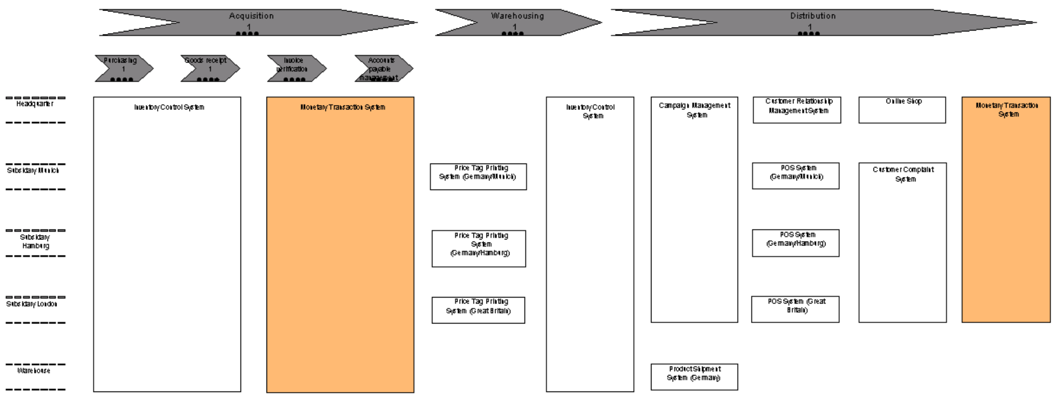


Figure 15.13.: *System Architect* - axis Landscape Management: Planned landscape

Although the information, which project affects which application systems, is stored within the repository, this information cannot be reused to automatically derive the planned landscape. Nevertheless, all visualizations mentioned above can be generated automatically leveraging the integration of Microsoft Visual Basic for Applications (VBA). Furthermore, impact analyses can be performed on the graphical models of the application landscape to identify differences between e. g. the current and planned landscape[9]. The results can either be displayed textually or graphically (see Figure 15.15).

Additionally, graphical models of the application landscape can be utilized to create different variants of the planned landscape and compare these variants leveraging the reporting capabilities of *System*

[9]For a detailed description of the impact analysis functionality of *System Architect* see Section 15.1.7.

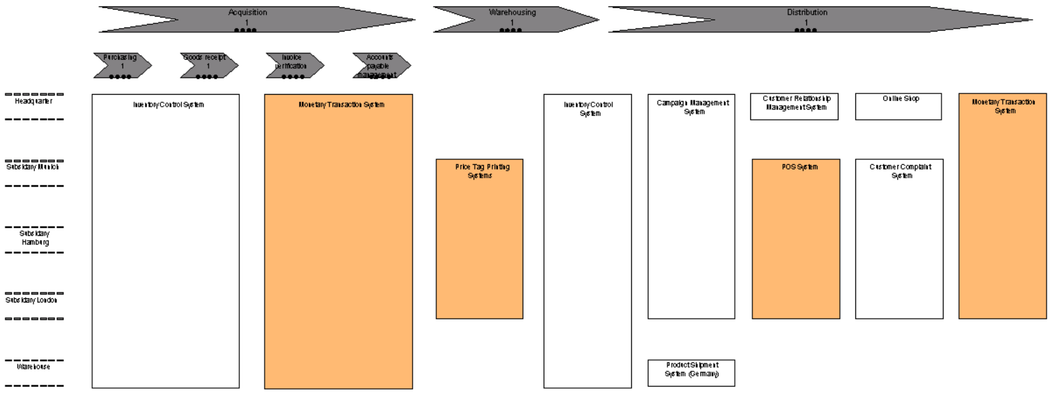


Figure 15.14.: *System Architect* - axis Landscape Management: Target landscape

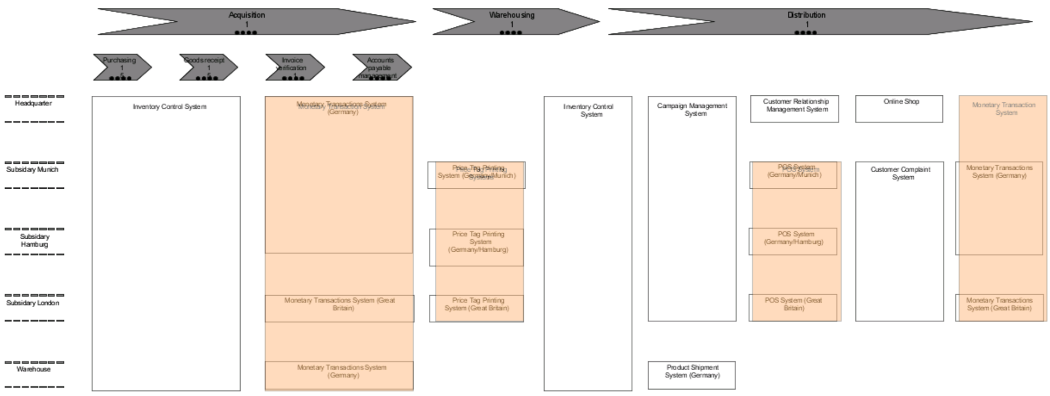


Figure 15.15.: *System Architect* - axis Landscape Management: Graphical comparison between current and target landscape

Architect. Furthermore, a history function is provided, which allows the auditing and tracking of changes. Thereby, information as performed change, modifying user, modification time is stored. Supplementary, the exporting functionalities e. g. as HTML file, can be used to historicize a certain status of the application landscape.

The procedure consistency is mostly given. Utilizing temporal information and graphical models to manage the evolution of the application landscape offers both flexibility and reusability. Nevertheless, the creation of the different visualizations demands manual effort as well as the definition of the reports used.

The procedure integration is mostly given. *System Architect* can reuse most of the data previously entered into the tool. However some information, e. g. the effects of projects regarding the evolution of the application landscape could not be reused to automatically derive future developments.

Rating: 5

15.2.2. Demand Management

The predefined information model of *System Architect* can be adapted to support the management of demands. Thereby, the different relationships concerning e. g. effects on application systems, business processes, or the derived projects can be specified. Accessing received demands can be performed either utilizing the *explorer* or creating a report, which lists all demands and selected attributes thereof (see Figure 15.16).

id	Name	Description	urgency
	Name		
40001	Improve Customer retention	The percentage of customer relationships that a business is able to maintain on a leng-term basis	high-middl
	Mr. Maier		
40002	Homogenization	Reducing the heterogeneity of business applications	high-middl
	Mrs. Huber		
40003	Reduce costs	Cost-efficient support and maintenance for business applications	high-middl
	Mr. Hofer		
40004	Adapt to government regulations	Understanding the government regulations for business applications	high-middl
	Mr. Schmidt		
40005	Improve enterprise-wide knowledge management	Building an intellectual capital strategy	high-middl
	Mrs. Huber		

Figure 15.16.: *System Architect* - axis Demand Management: Report listing all received demands and selected attributes

The deduction of projects derived from the given demands can be performed graphically leveraging the functionalities provided by the *explorer diagram*. Thus, demands are visualized on the respective diagram leveraging the drag and drop functionality provided by *System Architect*. Related projects can be automatically visualized utilizing the *show immediate relatives* functionality. An exemplary visualization displaying received demands and affected application systems is shown in Figure 15.17. Whereas, new demands and projects can be introduced graphically leveraging the visualization capabilities provided by *System Architect*, the relationship indicating which projects result from which demands can only be created graphically, if a definition type is used for implementing the relationship[10].

System Architect supports different possibilities to perform impact analyses concerning the potential effects of demands on other elements of the application landscape, e. g. application systems, organizational units, business processes. As mentioned before, a graphical analysis can be performed leveraging the capabilities provided by the *explorer diagram*. Another possibility is to utilize the reporting functionalities to create customized reports. Furthermore, this reporting capabilities can be used to identify similar demands to a selected one according to overlaps in affected elements from the repository, e. g. application systems.

The procedure consistency is mostly given. Demands and projects can be maintained using the functionalities provided by *System Architect* and the objectives could be created, although some manual effort was necessary to specify the required reports.

The procedure integration is completely given, as data previously entered in the tool can be reused in this context.

Rating: 5

15.2.3. Project Portfolio Management

Projects and portfolios thereof can be managed utilizing the functionality provided by *System Architect*. Although no built-in project portfolio management process is provided by the tool, due to the fact,

[10]Section 15.1.5 details on the different possibilities to introduce a relationship in the information model of *System Architect*.

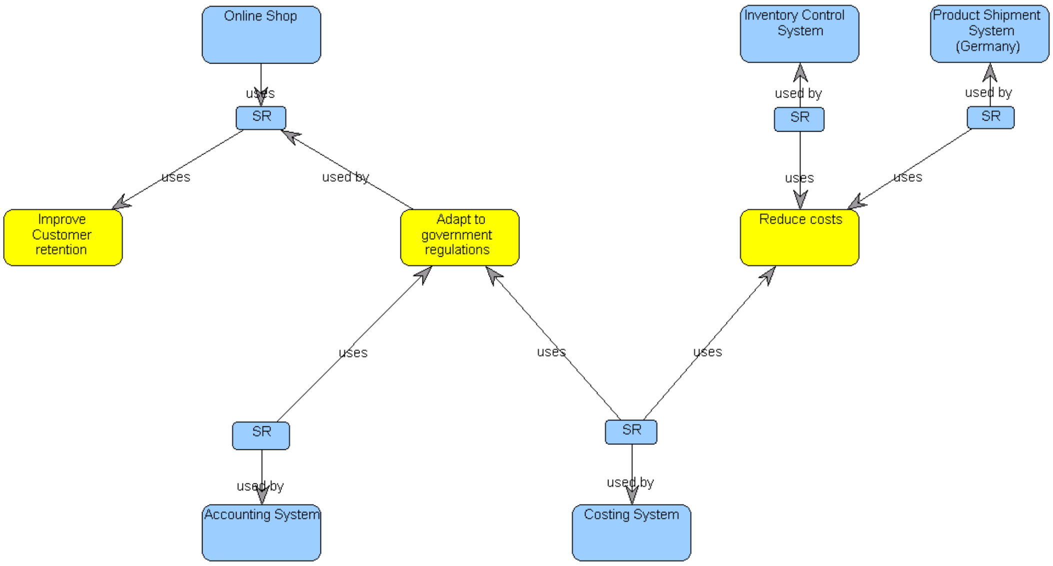


Figure 15.17.: *System Architect* - axis Demand Management: Graphlayout map illustrating received demands and affected application systems

that *System Architect* focuses on modeling the application landscape[11], the predefined information model supports the concept of project. Information on projects, e. g. start, end date, status, costs, and urgency, can be stored for each project complemented by relations between the project and other elements of the application landscape, e. g. demands, application systems, and organizational units. An overview of the received projects and selected attributes thereof, can be created e. g. leveraging the reporting functionalities as mentioned in Section 15.1.7.

As measures play an important role during the execution of project portfolio management, this information, e. g. ROI or project costs, can be maintained in *System Architect*. Although no direct support for the creation of portfolio matrices is provided by *System Architect*, the integrated functionalities of VBA can be leveraged to generate portfolio matrices displaying different financial aspects as the one shown in Figure 15.18. Reports illustrating financial calculations for a portfolio of projects are not directly supported by *System Architect*, but can be created either leveraging the built-in functionality of VBA or by utilizing the exporting functionalities to Microsoft Excel or *Telelogic Focal Point*.

Impact analyses to derive e. g. application system, business processes, or organizational units, affected by the execution of a project can be performed either graphically using the *explorer diagram* or by utilizing the *report generator*. An exemplary report specification listing all business processes, which are supported by an application system that will be changed by a project, is shown in Figure 15.19.

To identify potential conflicts between projects, due to e. g. the modification of the same application system, it is also possible to rely on the functionalities provided by the built-in *report generator*. The result of the execution of the report query is shown in Figure 15.20. Nevertheless, the reports described above are not shipped with the tool and need some time of familiarization with the information model to be built.

Due to the fact, that deriving the evolution of the application landscape is not directly supported by *System Architect*, graphical impact analyses on the application landscape as visualized in Figure 15.13 using color-coding cannot be performed automatically. Additionally, a distinction of *mandatory* and

[11] *Telelogic* provides a dedicated tool for project portfolio management, which is called *Telelogic Focal Point*.

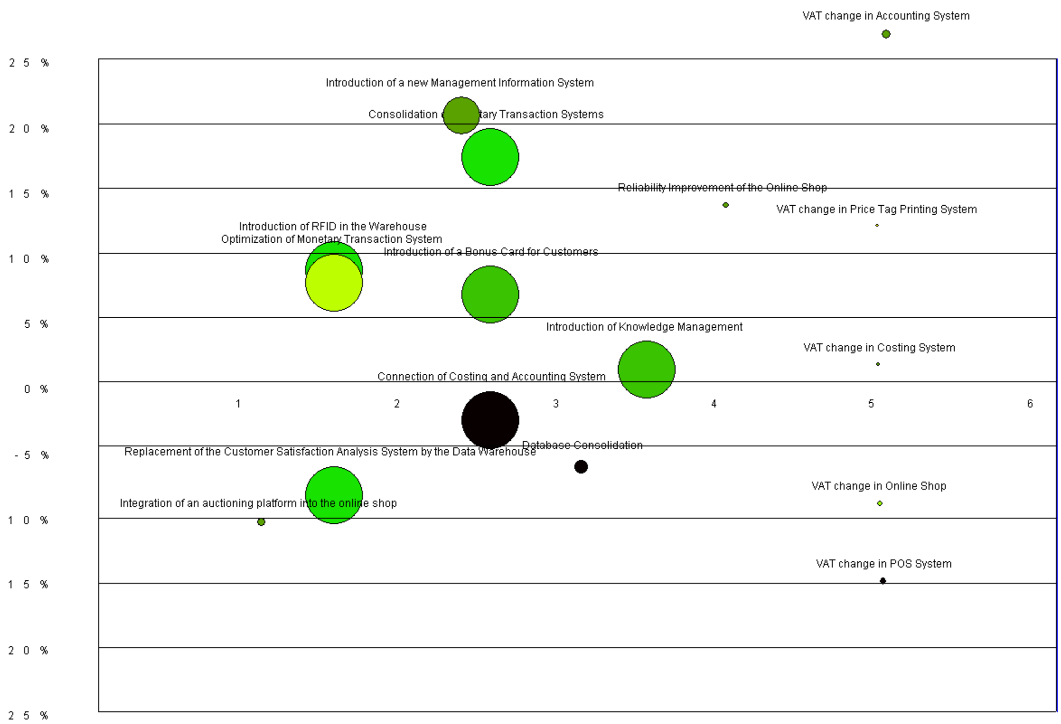


Figure 15.18.: *System Architect* - axis Project Portfolio Management: Portfolio matrix illustrating a project portfolio and selected financial aspects

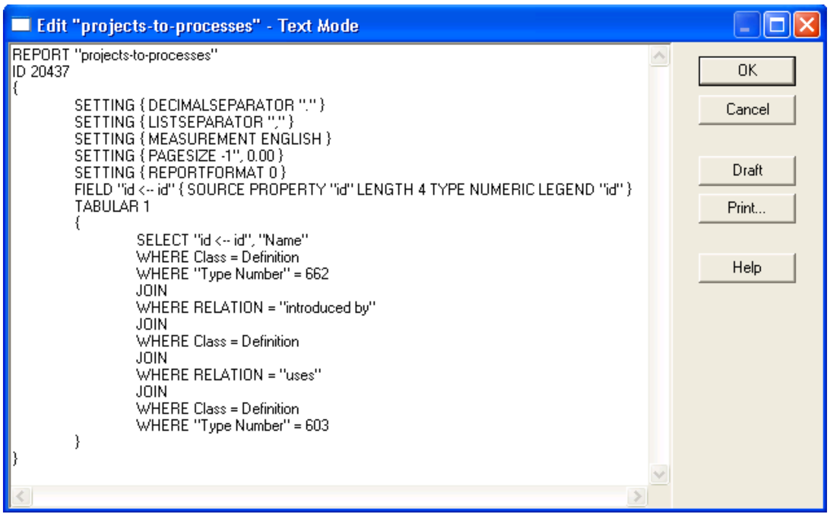


Figure 15.19.: *System Architect* - axis Project Portfolio Management: Definition of a report listing affected business processes

optional projects is not provided by the predefined information model, but can be introduced by adaptation.

Name	id	
	id	Name
Connection of Costing and Accounting System	12	
	18	Introduction of a new Management Information System
VAT change in Accounting System	7	
	12	Connection of Costing and Accounting System
VAT change in Costing System	8	
	18	Introduction of a new Management Information System

Figure 15.20.: *System Architect* - axis Project Portfolio Management: Report displaying potential conflicts between two projects

The procedure consistency is mostly given. Although *System Architect* provides support for most of the project portfolio management tasks, some tasks, e. g. project cost calculation or performing impact analyses on affected elements cannot be performed without customization effort.

The procedure integration is mostly given. Most of the data previously entered into *System Architect* can be reused in this context. Nevertheless, some information e. g. planned landscape scenarios could not be reused to derive the respective project portfolio.

Rating: 4

15.2.4. Synchronization Management

Synchronization aspects concerning multiple projects currently affecting the application landscape and their interdependencies can partially be addressed using *System Architect*. The predefined information model supports the storage of temporal information as start and completion date. Although different work items, their temporal information, and the status of their completion can be saved in *System Architect*, no direct support for the concept of project delay is provided. Nevertheless, such support can be realized by adapting the information model.

Whereas the visualization of temporal aspects of projects and the affected architectural elements, e. g. application systems, on diagrams is not directly supported by *System Architect*, a Gantt-like visualization can be created manually as shown in Figure 15.21[12]. To reduce the manual effort in creating and maintaining these diagrams, the integration of VBA can be leveraged to allow automated creation of the diagrams.

Leveraging the reporting functionalities provided by *System Architect*, potential project interdependencies according to an overlap in the affected architectural elements, e. g. organizational units, application systems, and business processes can be identified. In addition to textual reports, a graphical impact analysis can be performed. Utilizing the graphical facilities provided by *System Architect*, a user can create an *explorer diagram* containing the project under consideration and automatically displaying all *immediate relatives* thereof (see Figure 15.22). Thereby, the objects to be displayed can be filtered and highlighted utilizing the *explorer object report* or the *explorer relationship report* to improve readability.

Although, as mentioned above, the concept of project delays is not directly supported by *System Architect*, the flexible reporting functionality can be utilized to perform impact analyses regarding project delays. For instance, a report could be created that identifies projects, which are in danger of a project delay, as their completion date has passed by and the project itself is not specified to be completed. Nevertheless, the definition of this report demands manual adaption each time the report is executed as the current date must be defined manually.

[12]Telelogic provides an own product for product and project portfolio management called Focal Point, which supports among other functionalities the creation of time interval maps.

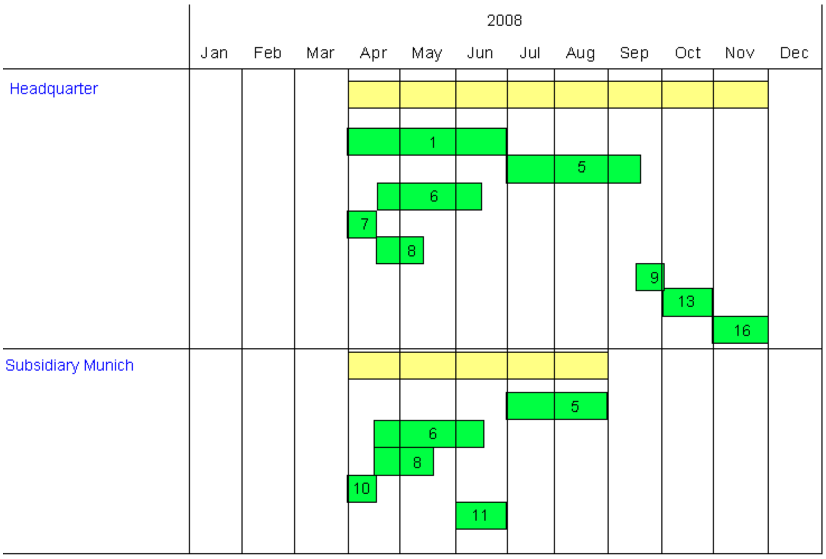


Figure 15.21.: *System Architect* - axis Synchronization Management: Time interval map visualizing the projects and the affected organizational units

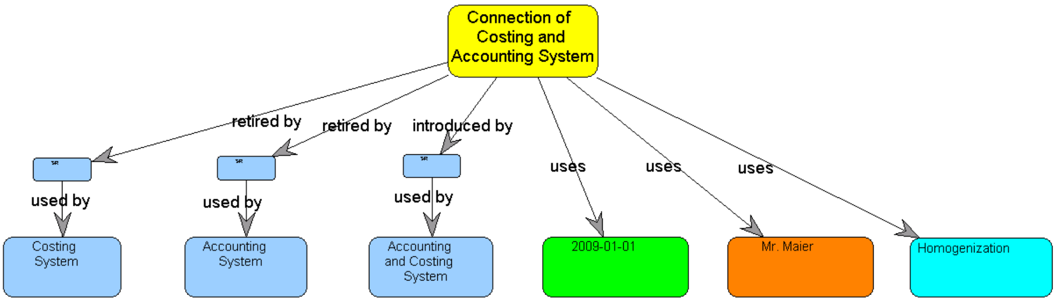


Figure 15.22.: *System Architect* - axis Synchronization Management: Explorer diagram illustrating the impact of a project on other elements

The procedure consistency is mostly given. *System Architect* provides basic support for synchronization aspects regarding multiple projects affecting the application landscape.

The procedure integration is mostly given. Information previously entered in the tool can be reused in this context. Nevertheless, the handling of temporal aspects requires some manual effort.

Rating: 4

15.2.5. Strategies and Goals Management

System Architect provides built-in support for strategies and goals management, as both concepts are shipped with the predefined information model but can additionally be adapted according to the specific needs of the user. Thereby, the predefined information model also supports the specification of metrics to measure the fulfillment of the respective goal. Accordingly, the relationship, which goal operationalizes which strategies, can be maintained as well as the demands and projects, which are executed to achieve the goal. These relationships can be created and maintained utilizing the *matrix browser* (cf. Figure 15.2).

Performing graphical impact analyses regarding the operationalization of a strategy by traversing from the respective strategy to affected elements of the application landscape can be performed utilizing an *explorer diagram*, as shown in Figure 15.23. This visualization can be created semi-automatically by creating a diagram containing the selected strategy and automatically including *immediate relatives* thereof.

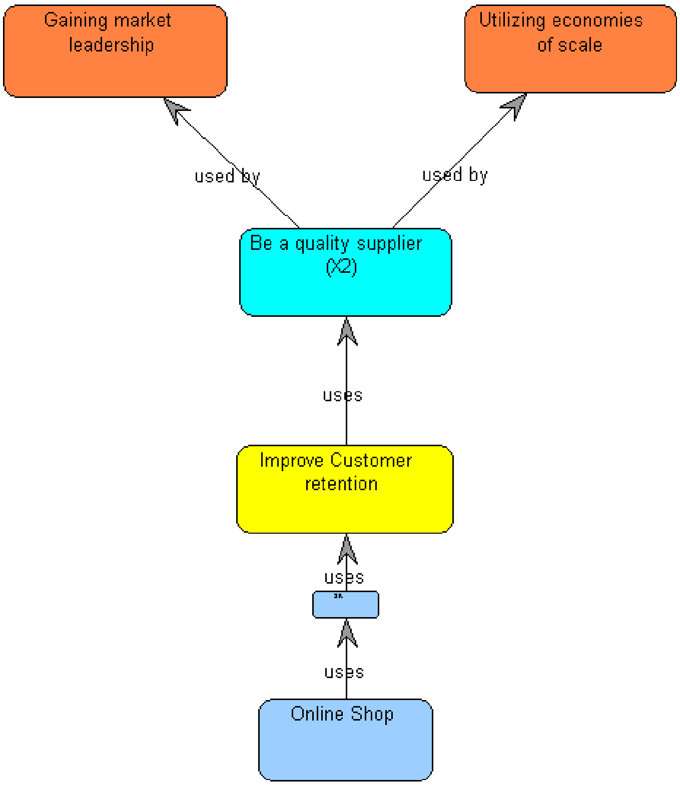


Figure 15.23.: *System Architect* - axis Strategies and Goals Management: Impact analysis starting with the strategies

Similarly, an impact analysis can be used to determine, which application systems are affected by which project that implements a goal operationalizing a selected strategy (cf. Figure15.24). Furthermore, a predefined diagram type of *System Architect* – the *strategy map* – can be used to perform graphical impact analyses. The makeup of the strategy map is similar to the one of a *balanced scorecard*. Thereby, the perspective types financial, customer, internal process, and employee are supported. Additionally, customized diagrams can be defined by adapting the information model (see Section 15.1.2 for details).

In order to evaluate, which goals owned by a specific organizational unit have been reached, *System Architect* provides the possibility to store the metrics used to measure the achievement of a goal. Nevertheless, if more sophisticated calculations concerning the goal achievement should be performed, the integrated VBA scripting functionalities can be utilized, which demands programming effort from a user. The result can be displayed e. g. using the provided HTML or Microsoft Word interfaces.

The procedure consistency is mostly given. The information concerning strategies and goals as well as the effects on other elements of the application landscape can be stored and maintained by the tool. Nevertheless, some of the objectives require a manual effort to be achieved.

The procedure integration is completely given. Data previously entered into *System Architect* can be reused in this context.

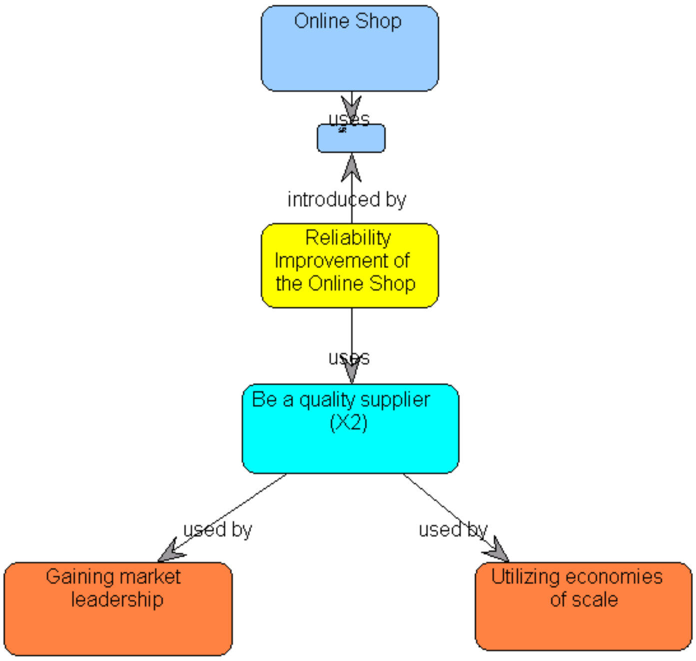


Figure 15.24.: *System Architect* - axis Strategies and Goals Management: Impact analysis starting with an application system

Rating: 6

15.2.6. Business Object Management

The predefined information model of *System Architect* provides support for business object management based on the class *Data Object*, which is suitable for representing business objects and can be renamed according to the terminology used in this survey. Additionally, the predefined class can be customized to include user-definable attributes. The business object class can be linked to corresponding data, e. g. an invoice template, as well as to other elements contained in the repository, e. g. application systems, which perform actions on the corresponding business object.

Providing an overview on the business objects that are exchanged between different application systems is possible by utilizing the matrix browser. If additional information, e. g. the operations performed on the business objects should be included, the *report generator* can be leveraged (cf. Figure 15.29). A similar custom report can be created that lists the information flow between application systems and the interface types thereby used (cf. Figure 15.25).

More sophisticated analyses, e. g. regarding the business objects required during the execution of a business process and their operation type (manual or automated), are not directly supported by *System Architect*. Such analyses can either be performed by utilizing the flexible visualization techniques (cf. Figure 15.26) of *System Architect* or by extending the provided functionality using the integrated functionality of VBA. In the latter case, a skilled user is required to specify the appropriate extension.

The procedure consistency is mostly given. Business objects, application systems, and interconnections can be analyzed leveraging the reporting functionalities to create customized reports or utilizing the manual visualization functionalities provided by *System Architect*.

The procedure integration is completely given. Information entered in previous scenarios can be reused in this context.

Name		
	type	
		Name
Accounting		
	offline	
		Inventory Control System
		Campaign Management System
		Campaign Management System
		Inventory Control System-1
		Accounting
		Inventory Control System
		Inventory Control System-2
Campaign Management System		
	manual	
		Inventory Control System
		Campaign Management System
		Campaign Management System
		Inventory Control System-1
		Inventory Control System
		Inventory Control System-2
	offline	
		Inventory Control System
		Campaign Management System
		Campaign Management System
		Inventory Control System-1
		Accounting
		Inventory Control System
		Inventory Control System-2
	online	
		Campaign Management System
		Customer Complaint System
		Campaign Management System

Figure 15.25.: *System Architect* - axis Business Object Management: Report listing information flows between application systems

Rating: 5

15.2.7. SOA Transformation

The SOA transformation scenario is supported in *System Architect* by the predefined information model, which can be additionally customized to include user-definable attribute like service level agreement (SLA). Following the hybrid approach to identify potential candidates for services, the reporting functionality provided by *System Architect* can be utilized to define custom reports containing the required information. In addition, graphical impact analyses and visualizations can be leveraged to illustrate the impact of the transformation on the application landscape. *Telelogic* provides a pur-

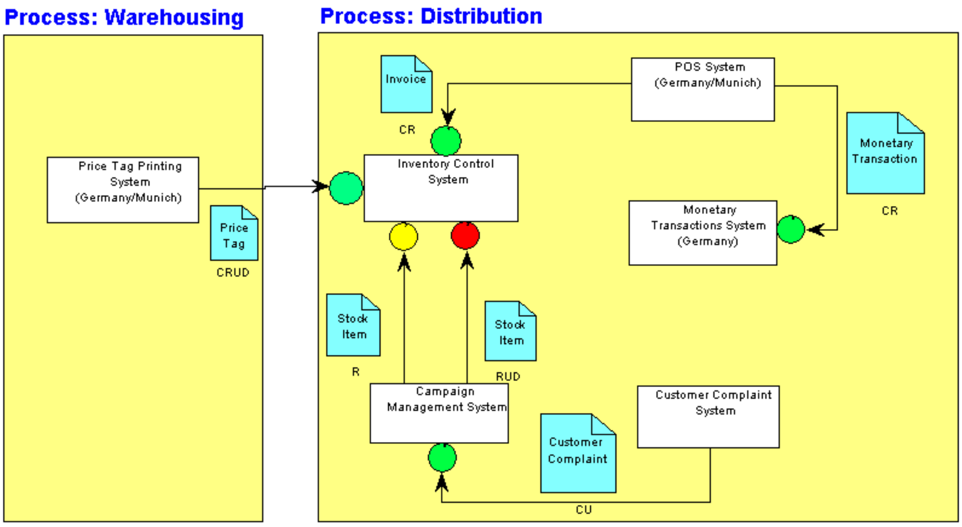


Figure 15.26.: *System Architect* - axis Business Object Management: Cluster map illustrating information flows between application systems

chasable add-on, which provides a customized information model and predefined reports called *SOA solution*, which can be leveraged to reduce the manual customization effort[13].

Leveraging the top-down approach and trying to identify potential candidates for services according to the usage in the execution of a business process, a graphical representation of the application landscape can be used. Figure 15.27 for instance, highlighting differentiating business processes. Thereby, the positioning and highlighting of the visualization has to be performed manually or could be automated utilizing the integrated VBA scripting engine.

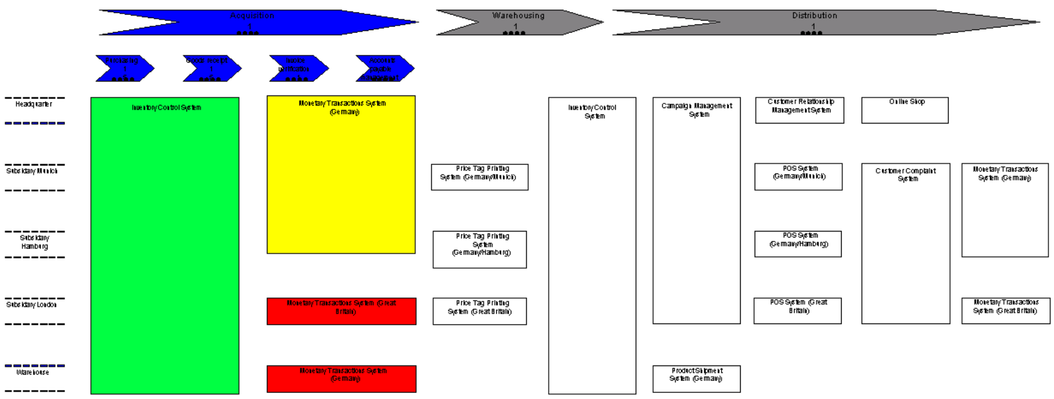


Figure 15.27.: *System Architect* - axis SOA Transformation: Process support map visualizing differentiating business processes and the frequency of changes of application systems using color-coding

[13]Section 15.1.5 details on the possibilities provided by these paid add-ons.

The visualization depicted in Figure 15.27 can also be used to answer more technical questions, e. g. which application systems change frequently. Combining the information about the frequency of change regarding an application system with information about which information systems should be modified in the near future may help to decide which application systems are potential candidates for a transformation into a service. Figure 15.28 provides the result of a user-defined report listing application systems that will be modified in the next year.

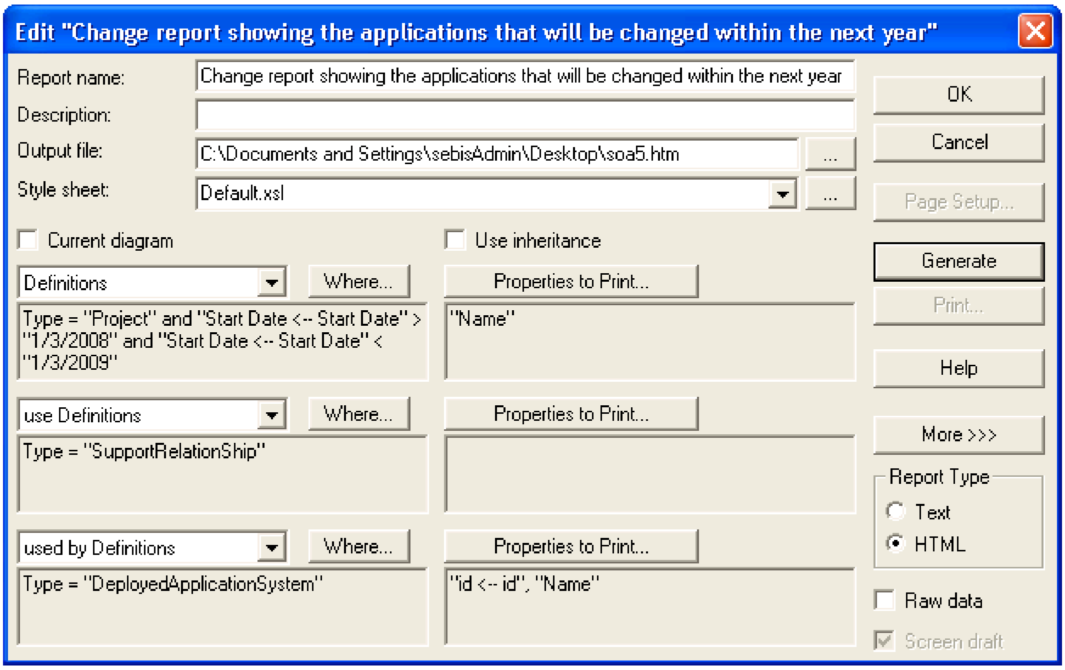


Figure 15.28.: *System Architect* - axis SOA Transformation: Report listing application systems that will be modified in the next year

Similar anlyses can be performed leveraging the *explorer diagram*, e. g. to depict application systems that are used in numerous domains, or the *report generator* to define custom reports, e. g. to list business objects and the operations performed on them (cf. Figure 15.29).

In order to illustrate the impact of the SOA transformation on the application landscape, *System Architect* provides varying possibilities, ranging from graphical analyses to user-definable reports. In either case, a manual effort is required to perform the positioning and highlighting of elements on the visualization or to specify the required report. In order to reduce this effort the *SOA solution*, a paid add-on, provided by *Telelogic* can be utilized.

The procedure consistency is mostly given. *System Architect* supports the creation of most of the objectives specified in the scenario description, although a certain manual effort is required from the user.

The procedure integration is completely given, as data previously entered and reports created in preceeding scenarios can be reused to simulate the scenario.

Rating: 5

ApplicationSystem		
	BusinessObject	type
Accounting		
	Customer	RU
Campaign Management System		
	Stock Item	R
	Customer	RU
	Stock Item	R
	Customer	R
	Stock Item	R
	Customer	RU
	Stock Item	R
	Customer	RU
Campaign Management System		
	Stock Item	R
	Customer	RU
	Stock Item	R
	Customer	R
	Stock Item	R
	Customer	RU
	Stock Item	R
	Customer	RU

Figure 15.29.: *System Architect* - axis SOA Transformation: Report listing business objects and the operations performed on them

15.2.8. IT Architecture Management

The predefined information model of *System Architect* does not provide concepts to describe abstract architectures for application systems like e. g. client-server or 4-tier-thin-client-architecture. Nevertheless, leveraging the possibilities to adapt the predefined information model, concepts like *architectural blueprints, architectural solutions, blueprint elements,* and *solution elements* can be introduced as well as the relationships described in the information model of *SoCaStore* (see Section 3.2).

Having introduced the concepts as mentioned above, the different possibilities to perform impact analyses as provided by *System Architect* can be used. Figure 15.30 depicts the result of a custom report listing application systems, the architectural solution they should conform to, and the solution elements the application system is based on.

Furthermore, graphical impact analyses can be conducted, e. g. displaying an application system, the architectural solution it should conform to, and the solution elements the architectural solution is constituted of, can be generated utilizing the *explorer diagram* functionality. Another visualization illustrating a holistic view on the application landscape and the conformance of the application systems to their architectural solution using color-coding is depicted in Figure 15.31. Thereby, the positioning and highlighting has been done manually but can be automated leveraging the integrated scripting functionalities of VBA.

id	Name	shouldConformTo ArchitecturalSolution
500	Accounting System	OracleTomcat
	Tomcat 5.1	
	Apache 2.0.53	
	IE 6.0	
	Oracle 9i	
510	Accounting System	OracleTomcat
	Tomcat 5.1	
	Apache 2.0.53	
	IE 6.0	
	Oracle 9i	
520	Accounting System	OracleTomcat
	Tomcat 5.1	
	Apache 2.0.53	
	IE 6.0	
	Oracle 9i	
600	Costing System	MySQLBeaFatClient
	MySQL 2.1	
	Bea Weblogic 8.1	
	Proprietary Fat-Client	

Figure 15.30.: *System Architect* - axis IT Architecture Management: Report listing application systems, architectural solutions they should conform to, and the solution elements they are based on

Figure 15.31.: *System Architect* - axis IT Architecture Management: Cluster map visualizing application systems and their conformance to architectural solutions

Bar charts visualizing the number of usages of an architectural solution or of the solution elements are not directly supported by *System Architect*. Nevertheless, such visualizations can be created leveraging the integration of VBA scripting engine. The creation of the query and the visualization would require programming effort from a skilled user.

The procedure consistency is partially given. Although the concepts used in this scenario are mostly not supported by the predefined information model of *System Architect*, the information model can be adapted and most of the objectives can be achieved.

The procedure integration is completely given. Data previously entered into the tool can be reused in the context of this scenario.

Rating: 6

15.2.9. Infrastructure Management

The predefined information model of *System Architect* provides built-in support for different infrastructure elements, e. g. *database management systems* or *technologies*, but does not provide a general *infrastructure service* concept. In addition, the predefined information model lacks support for temporal information, e. g. regarding support times, like start of production, and aspects regarding costs. Therefore, the information model of *System Architect* has to be adapted. The scenario simulation as detailed below focuses on the usage of database management systems but similar considerations could be taken for other infrastructure services.

To get a first overview on the database management systems that are currently used in the application landscape of *SoCaStore*, a custom report can be created utilizing the *report generator*, which lists all solution elements of the type "database management system". In order to perform impact analyses e. g. to identify the database management systems, which are running out of support, the *report generator* can be leveraged to create custom reports addressing the specific questions. Figure 15.32 shows exemplary results of such a report.

id	Name	startIntroduction	endIntroduction	startProduction	endProduction	startPhaseOut	endPhaseOut	supportEndDate
240001	MySQL 2.1	1/1/2005	9/30/2005	10/1/2005	6/15/2009	6/16/2009	6/1/2010	12/31/2010
240005	Oracle 9i	4/1/2005	12/31/2005	1/1/2006	12/31/2013	1/1/2014	12/31/2014	1/31/2015
240008	DB2 6.0	6/1/2004	5/31/2005	6/1/2005	12/31/2009	1/1/2010	12/31/2010	6/30/2011

Figure 15.32.: *System Architect* - axis Infrastructure Management: Report listing database management systems that are running out of support

More sophisticated impact analyses, which additionally consider the application systems based on a database management system running out of support as well as the organizational unit, which hosts the respective application system can be performed graphically. Figure 15.33 gives an overview on the application landscape using a cluster map, which illustrates database management systems that are running out of support by color-coding. Thereby, the positioning and highlighting of the visualized symbols has to be performed manually or using *analytic depictions*[14] but could be automated leveraging the built-in functionalities of VBA.

Whereas different cost types and their values can be stored by adapting the information model of *System Architect*, calculating cumulative costs using the *report generator* is not directly supported. In order to perform such calculations the integrated VBA has to be utilized to create custom reports. Such reports could additionally be used to identify potential candidates for replacement during a consolidation of the database management systems.

[14]The concept of *analytic depictions* is detailed in Section 15.1.3.

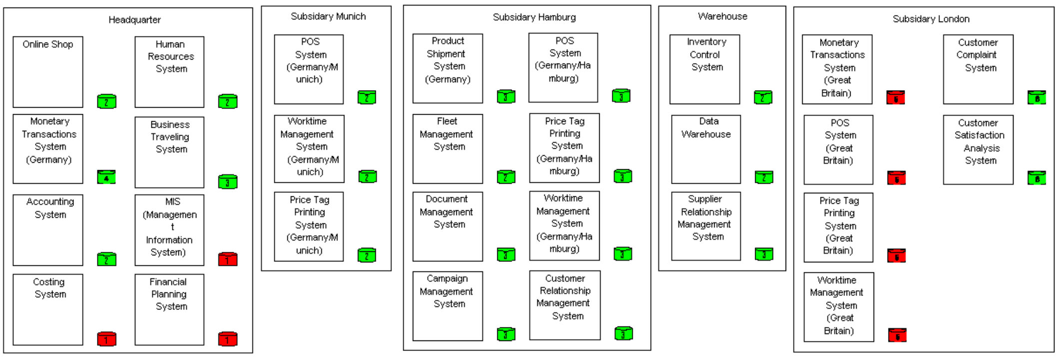


Figure 15.33.: *System Architect* - axis Infrastructure Management: Cluster map showing database management systems, which are running out of support by color-coding

To perform impact analyses to identify, which application systems are affected by the replacement of a database management system the graphical capabilities provided by *System Architect* can be leveraged. Thereby, either the *explorer diagram* can be used, which can be generated semi-automatically but does not provide a holistic view of the application landscape or a user-defined diagram, e. g. a cluster map, which has to be created manually. Figure 15.34 shows a cluster map visualizing application systems that are affected by the consolidation via color-coding.

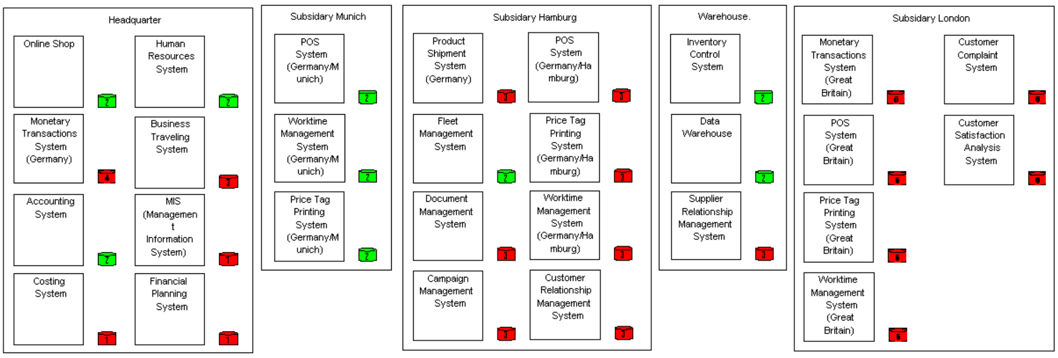


Figure 15.34.: *System Architect* - axis Infrastructure Management: Cluster map illustrating the application systems that are affected by the database management system consolidation

The procedure consistency is mostly given. The information model provided by *System Architect* has to be adapted to e. g. support lifecycle information of database management systems. Additionally, some of the objectives require manual effort to be achieved.

The procedure integration is completely given, as data previously entered into the tool can be reused for the simulation of the scenario.

Rating: 5

15.3. Tool Vendor's Profile

Hubert Zenner, Product Line Expert System Architect

Telelogic® is a leading global provider of solutions for automating and supporting best practices across the enterprise - from the powerful modeling of business processes and enterprise architectures to the requirements-driven development of advanced systems and software. Telelogic's solutions enable organizations to align products, systems, and software development lifecycles with business objectives and customer needs to dramatically improve quality and predictability, while significantly reducing time-to-market and overall costs.

To better enable our customers' drive towards an automated lifecycle process, Telelogic supports an open architecture and the use of standardized languages. As an industry leader and technology visionary, Telelogic is actively involved in shaping the future of enterprise architecture, application lifecycle management, and customer needs management by participating in industry organizations such as INCOSE, OMG, The Open Group, Eclipse, ETSI, ITU-T, the TeleManagement Forum and AUTOSAR.

Headquartered in Malmö, Sweden, with U.S. headquarters in Irvine, California, Telelogic has operations in 20 countries worldwide. Customers include Airbus, Alcatel, BAE SYSTEMS, BMW, Boeing, Daimler, Deutsche Bank, Ericsson, General Electric, General Motors, Lockheed Martin, Motorola, NEC, Philips, Samsung, Siemens, Sprint, Thales and Vodafone.

Leveraging Technology to Strengthen Your Competitive Advantage

Successful businesses leverage technology into competitive advantage by continuously improving current operations while innovating, evolving and creating new products and services. To make this a reality, the business managers and technologists in your organization must think, create, and adapt collaboratively.

Enterprise Lifecycle Management (ELM) enables you to align your technology investments with your enterprise objectives by automating and integrating:

BPO - Maximize the performance of your technology investments with Business Process Optimization (BPO): Manage your business processes; rapidly respond to marketplace change; eliminate wasteful business activity.

- Develop Enterprise Architecture integrating four key domains: Business, Information, Systems, and Technology in a single repository
- Model, simulate, and improve key business processes
- Manage your portfolio of products to maximize value, improve decision making and support the enterprise strategy

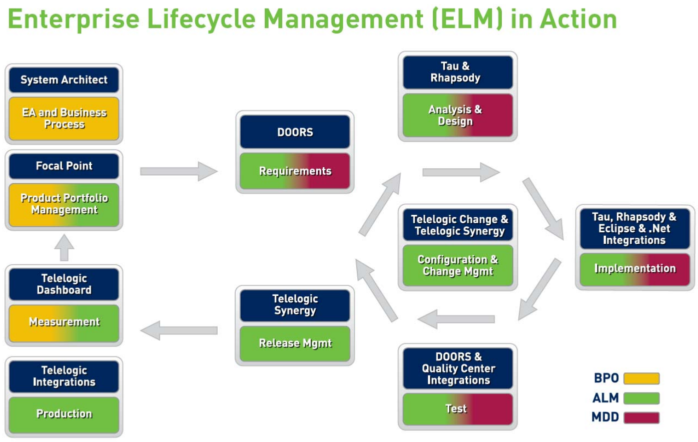
ALM - Improve the entire development lifecycle with Application Lifecycle Management (ALM): Increase productivity; develop innovative solutions; prove compliance.

- Ensure the voice of the customer remains persistent throughout the systems and software development process
- Improve collaboration, productivity and traceability across globally distributed teams without adding unnecessary overhead
- Assess the impact of, react to, and manage change

MDD - Manage system complexity with Model Driven Development (MDD): Reduce development time; improve productivity and quality; lower project risk.

- Visually develop systems and software

- Improve reuse and quality by automatically generating applications, documentation and tests
- Optimize productivity through iterative development



Telelogic System Architect® is a comprehensive enterprise architecture solution that enables you to design, visualize and analyze business models and enterprise architectures. You can see the relationships between technology, processes and data - as well as trace back these diverse business requirements to their original sources. As a result, you can more quickly understand, analyze and take firm action on the ever-changing technology and business issues that affect your organization. The tool family includes a Web-based enterprise architecture solution that transforms your distributed team into a unified workforce. Analysis templates provide multiple views into the organization - Strategic, Operational, and Technical - and focus on the information needed to promote effective decision-making at each level.

Telelogic Focal Point™ is a product portfolio management solution that improves product value and quality, from idea conception through the product lifecycle, with a continual focus on achieving customer satisfaction. The tool family includes a Web-based solution for market-driven product managers that helps you select the right product features and deliver them to the right market at the right time. With Focal Point, you select the products and features customers want by capturing and leveraging critical product and market information. By focusing on customer and business value, Focal Point improves the product decisions that you make throughout the lifecycle.

Telelogic DOORS® increases the quality of systems engineering and business-critical IT projects with powerful capabilities for capturing, linking, analyzing, and managing changes to requirements and their traceability. The tool family includes a quick-to-learn and easy-to-use solution for requirements definition and management that helps software development projects with short delivery timescales improve collaboration, meet business needs, and deliver quality applications rapidly.

Model Driven Development™ (MDD™) products increase development efficiency by creating design models that can be simulated and tested before automatically generating application code, documentation, and production test scenarios.

Telelogic Rhapsody® enables model driven development of complex embedded systems and software for embedded and/or real time systems.

Telelogic Tau® enables model driven development of complex systems and software for enterprise applications, including those utilizing Service Oriented Architectures (SOA).

Telelogic MDD solutions work seamlessly with IDEs like Eclipse, Green Hills MULTI, Wind River Tornado and Microsoft Visual Studio .NET.

CHAPTER 16

Troux Technologies Inc. (Troux 7)

Contents

The *Troux 7* is an environment with integrated technologies supporting different tasks as shown in Figure 16.1.

Figure 16.1.: *Troux 7* provided by *Troux*

Troux Metaverse is the repository containing the central information model called *Troux Semantics* and the central objects with their attributes and relationships.

The *Troux Architect* is the client for creating visualizations and changing data.

The *Troux Explorer* is the web-based J2EE compliant portal, which is customizable through the *Troux Composer* and therefore can be adapted based on the customers needs[1].

Troux Intelligence is the tool providing reports.

16.1. Evaluation of Specific Functionality

This chapter outlines the detailed results of simulating the scenarios from Section 4.1 with the *Troux* version 7.0.

[1]It consists of iFrames which can be embedded into corporate portals.

16.1.1. Importing, Editing, and Validating

The Microsoft Excel file can be completely imported to *Troux Metaverse* through the *Troux Explorer*. The import is realized in three steps. First the information model shipped with *Troux* has to be adapted to represent the information model of *SoCaStore*. Second, the importing script has to be adapted to conform the given Excel File. Third, the data from the Microsoft Excel file has to be imported by a function called the *Troux Collection*. These *collector* scripts are XML files, which can be created by hand or by creating a model of a collector within *Troux Architect* from which the collector script can be automatically generated. Leveraging this scripting functionality, an import procedure can be crated, which allows a flexible importing based on the given Microsoft Excel file including data transformation. Usually, the importing is realized by *Troux* consultants, but the scripting language could be leveraged by a skilled user as well.

The import functionalities provide comprehensive support for validation of consistence and data quality. During the import the status for extract, transform, and loading is shown, as presented in Figure 16.2. The status distinguishes between *Info*, *Warning*, *Error*, and *Catastrophic*. *Warning* e. g. means missing attributes which are not mandatory, while *Error* stands for data errors which need to be corrected, but which do not stop the import, e. g. missing attributes that are mandatory, and *Catastrophic* means that the import can not be continued, e. g. if an object should be imported that is not represented by a class in the information model. After the import the report results are stored and can be looked at.

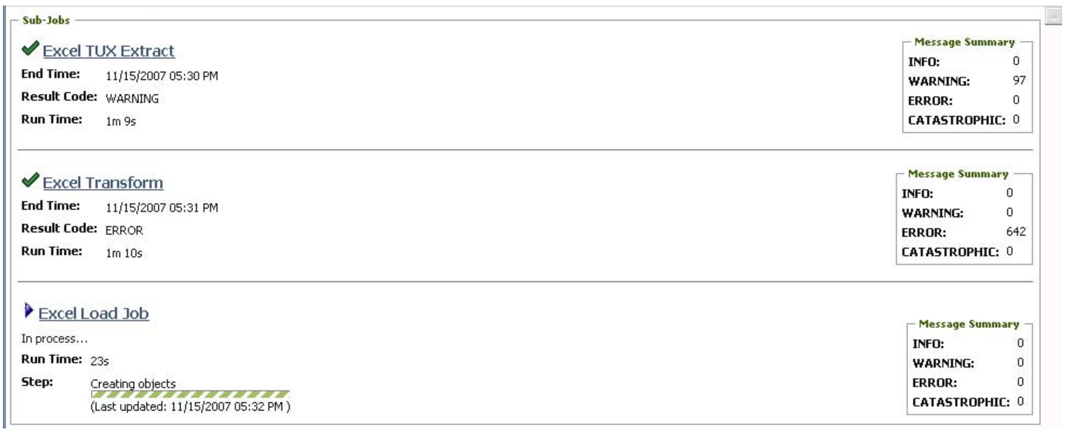


Figure 16.2.: *Troux* - axis Importing, Editing, and Validating: Importing Process (extract)

The quality of the data can be further checked, for example to see improvements regarding the data quality, e. g. if an manual clearing process occurs. A service process can be scheduled, which compares the data with the information model, and delivers a data quality report storing the results like shown in Figure 16.3. The report can be drilled down highlighting the absence of data for erroneous objects.

Once the data is stored in *Troux Metaverse*, it can be edited in several ways. First, it can be edited using the *Troux Architect* client. Figure 16.4 shows an example of editing application systems, where the application systems were placed into a model and their properties were shown. Changes to the data can be made, also multiple choices conveniently by picklists. Later, the changes made, can be pushed to *Troux Metaverse*.

Second, data can be edited through the web-based *Troux Explorer*. Figure 16.5 shows an example of editing the Working Management System (Germany/Hamburg) through *Troux Explorer*.

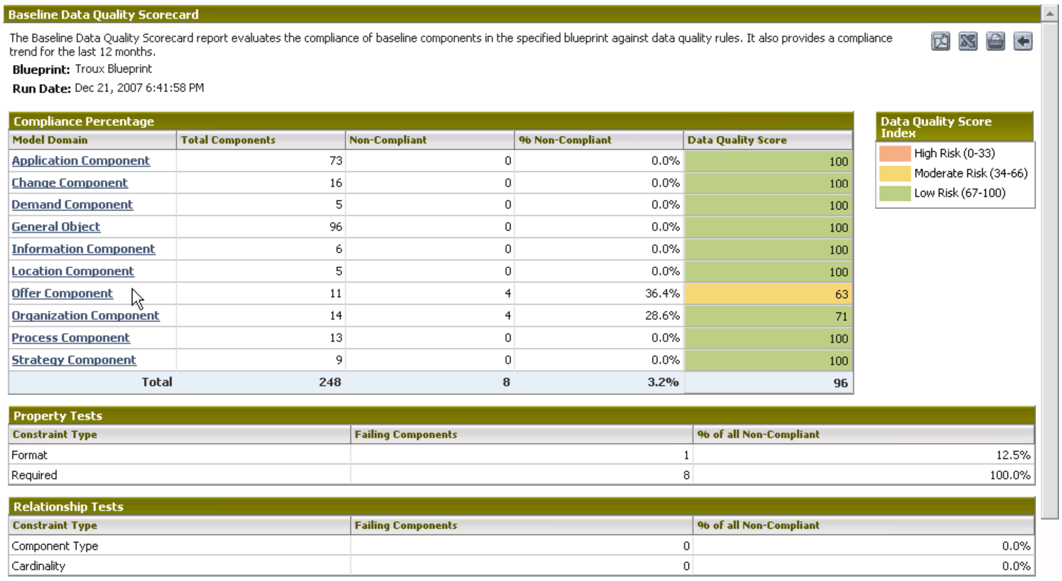


Figure 16.3.: *Troux* - axis Importing, Editing, and Validating: Report created by the ETL process

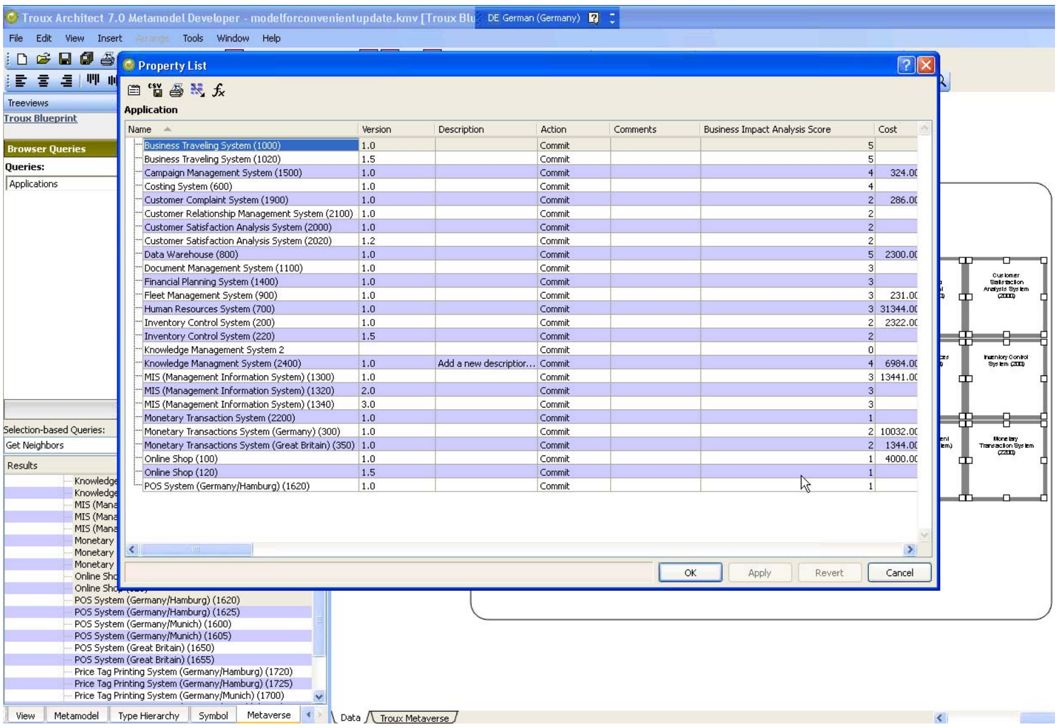


Figure 16.4.: *Troux* - axis Importing, Editing, and Validating: Editing application system properties through *Troux Architect*

Third, data can be edited in several other applications using structured data sources, like XML or CSV. Therefore, data can be exported to XML, or MS Excel, and is also available in standard database

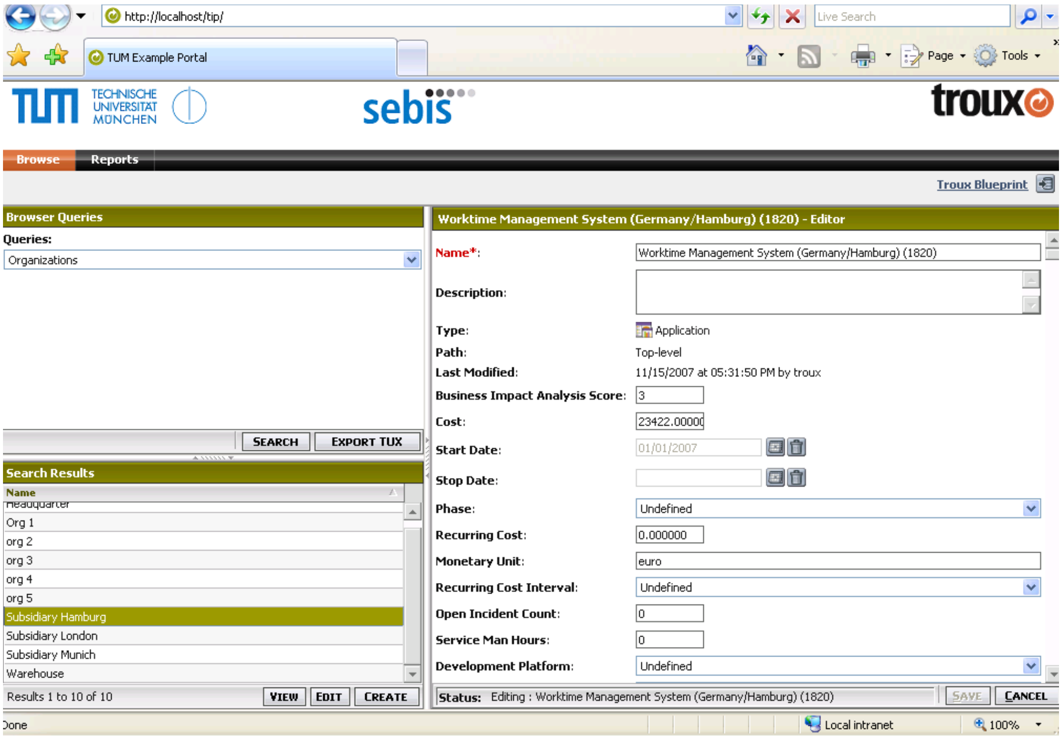


Figure 16.5.: *Troux* - axis Importing, Editing, and Validating: Manually Editing of the Working Management System (Germany/Hamburg) through *Troux Explorer*

tables[2]. The import of the data is realized through the *Troux Collection* on a scheduled basis. The *Troux Collection* has an XML configuration file, which specifies the location and type of the source, a query defining the subset of data desired from the source, and how to transform and map the data to the information model.

If data is edited outside and imported using the *Troux Collection* into *Troux Metaverse* the differences between the re-collected data and existing repository contents are updated. However, it will not be checked, if the data on *Troux Metaverse* was changed in the meantime and those changes will be discarded, by updating the data through the *Troux Collection*. However, all changes to *Troux Metaverse* content is logged as to what changed, the date and time, the previous value, and the user or *Troux Collection* id that made the change.

Rating: 5

16.1.2. Creating Visualizations

The *Troux Architect* is technically matured regarding the manual drawing functionalities. Therefore, an experienced user can create different visualizations quite fast, for example through the usage of shortcut keys or action buttons, that supply helpful drawing features for visualized objects, like left or right edge alignment, vertically and horizontally distribution, or force equal size. The following maps can be created manually *process support map*, *graphlayout map* (c.f. Figure 16.6), and *swimlane diagram* like shown in Figure 16.7.

[2]It is available in standard database tables for supported databases or business intelligence datamarts.

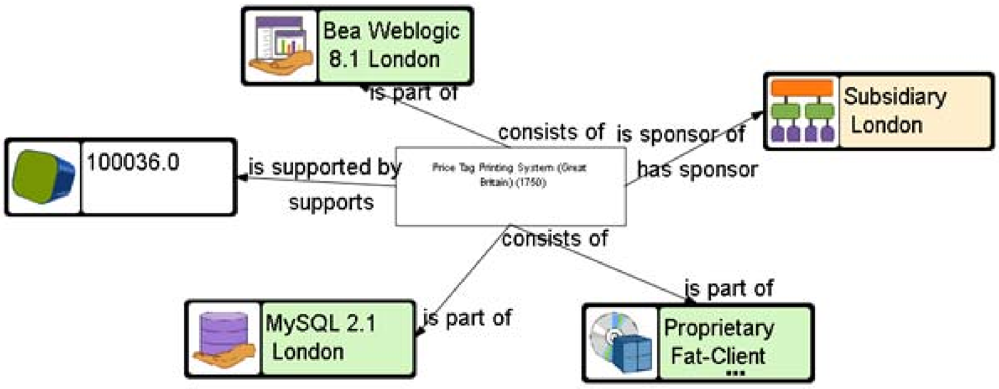


Figure 16.6.: *Troux* - axis Creating Visualizations: Graphlayout Map

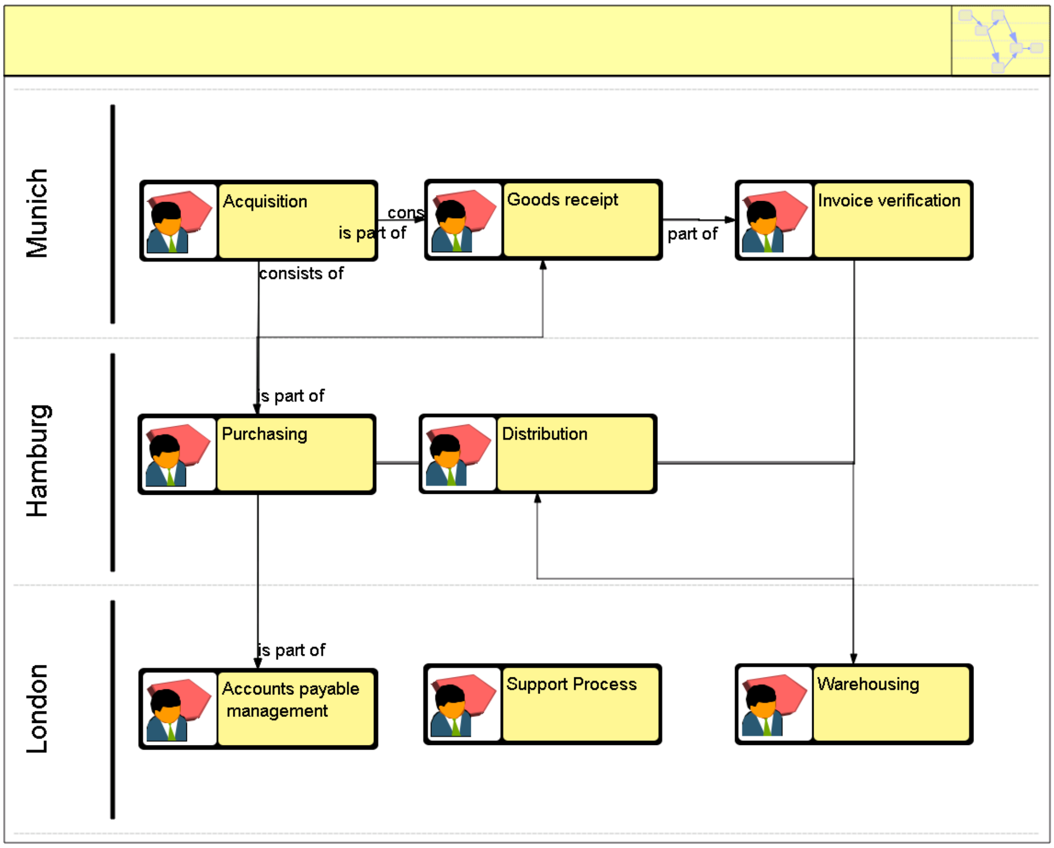


Figure 16.7.: *Troux* - axis Creating Visualizations: Swimlane Diagram

A *cluster map* (c.f. Figure 16.8) can be created as well. Therefore, *container* can be created that are saved with the visualization and are not part of the information model and therefore are not transferred to *Troux Metaverse*. All visualizations can be adapted to the needs of the customer, e. g. a visualization can support the representation of services instead of application systems.

A *time interval map* displaying an overview on the lifecycle information of e. g. an application system cannot directly be created. However, there exists the concept of lifecycles to store temporal information, therefore it is possibly to execute analyses taking those temporal aspects into account.

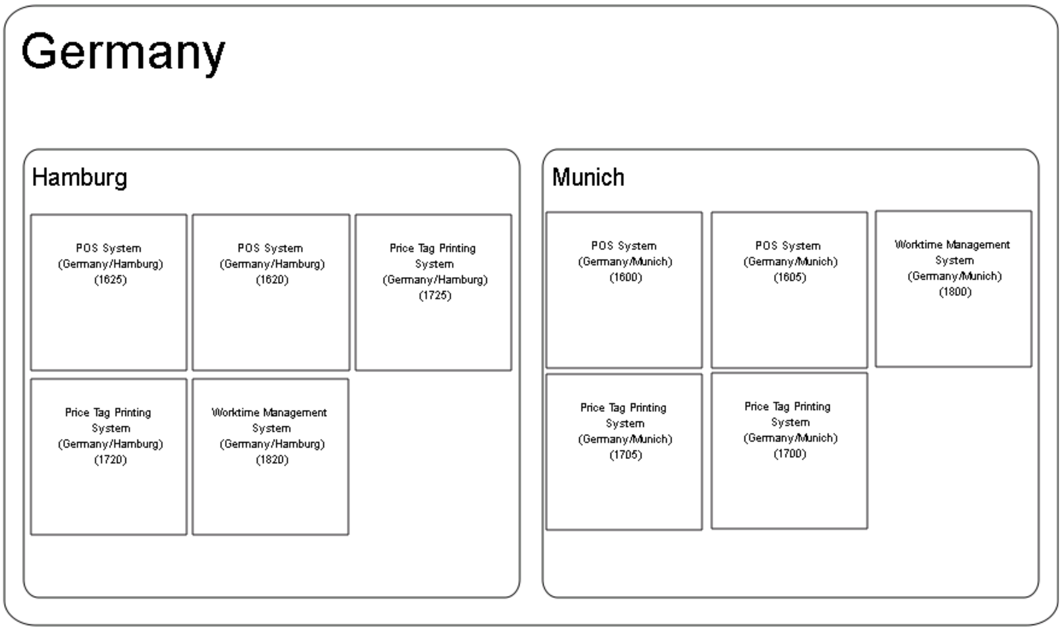


Figure 16.8.: *Troux* - axis Creating Visualizations: Application systems in Germany

Automatically generated software maps do not come out of the box with the *Troux Architect* but can be created by *Troux* based on customer needs. One example that can be realized shows Figure 16.9[3] Automatically created visualizations can be adapted by the customer, however a deep understanding of *Troux Architect* and its functionality is required.

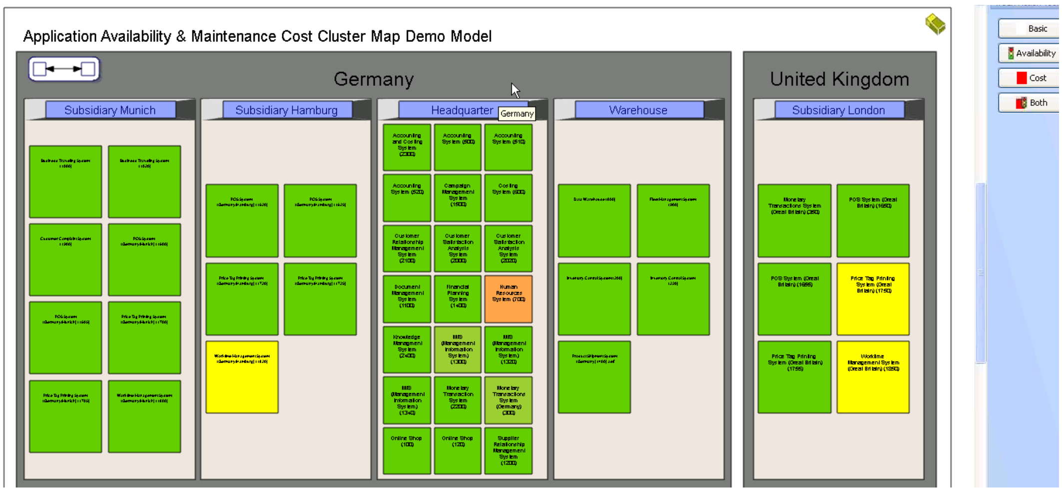


Figure 16.9.: *Troux* - axis Creating Visualizations: Application systems: Maintenance Costs

If the data set changes, automatically created visualizations can be updated. However, manual changes in the visualization that did not lead to a semantic change, e. g. if a certain application system should always be in the upper left corner, are disregarded.

[3]The template was created in cooperation by act! and troux consulting.

Troux Architect provides multiple features for manual drawing of visualizations and a skilled user can create visualizations quite fast. However, for an untrained user vocational adjustment time or doing a training first is recommended. For example the size of boxes within a diagram should always be resized per action buttons and not per drag & drop. Drag & drop is available, but performing it this way, can lead to objectionable results, if the diagram is opened by another *Troux Architect* client.

Rating: 4

16.1.3. Interacting with, Editing of, and Annotating Visualizations

The *Troux Architect* offers the capability for making changes, including deletions, of data offline. Thereby, only adaptations which are conform with the information model are allowed. Later the user can decide, if the changes should be synchronized with *Troux Metaverse*. If changes to a visualization are made, like adding new objects, relationships, or changing attributes, those edits can be pushed to *Troux Metaverse* like shown in Figure 16.10. Also automatically created visualizations can be refreshed on demand with the newest data from *Troux Metaverse*.

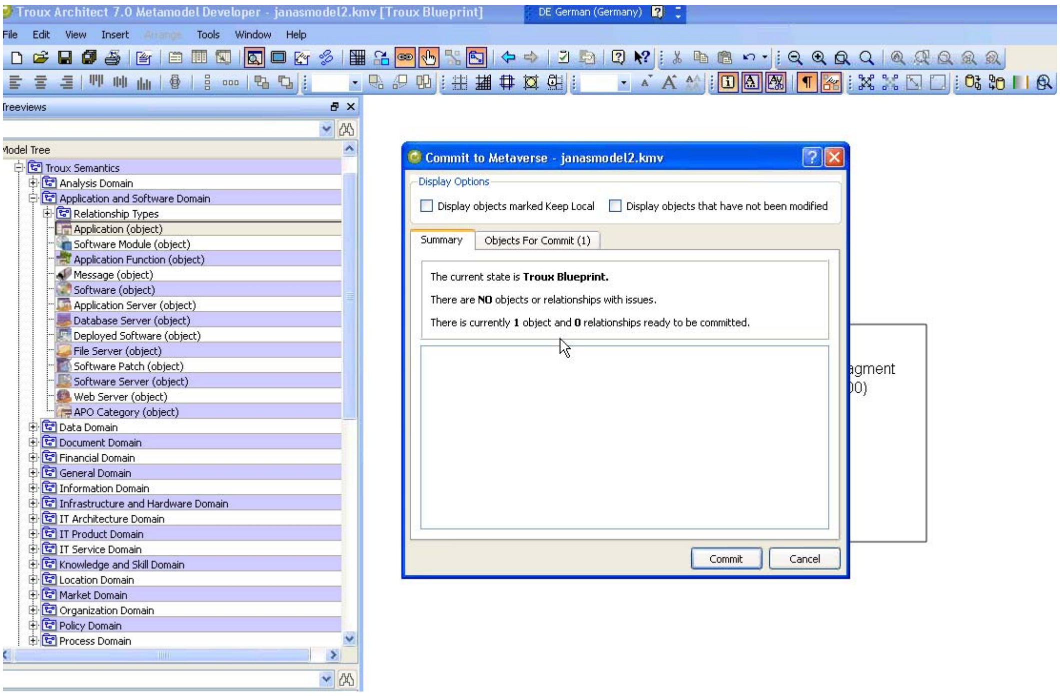


Figure 16.10.: *Troux* - axis Interacting with, Editing of, and Annotating Visualizations: Committing changes in the data from *Troux Architect* to *Troux Metaverse*

Troux Architect provides several filtering and search mechanisms for visualizations. Objects and relationships can be hidden in order to highlight certain elements or to reduce the complexity of visualizations. For example, the following hiding features are provided by clicking action icons:

- hide / show all relationships
- show relationships only to and from selected objects
- inversion of selected objects

Furthermore, all containers[4] and sub-containers can be opened and closed by double-click. This feature is not only useful for hiding of complex inner structures, but also all objects in closed containers will not be loaded to *Troux Architect*. This allows a fast loading of visualizations and is especially useful for large scale data.

Interactive search capabilities and navigation mechanisms are provided, e. g. all neighbors of an application system can be highlighted. Also, within an application system, traversing from a selected element to a connected element is possible.

Furthermore, once a basic visualization is created, it can be annotated with additional aspects, like background colors or additional symbols. Figure 16.11 shows an example where the background color and the traffic light symbol next to each application system depend on certain attributes, like availability or maintenance costs, of the application systems itself. If the attributes in the data set change, the visualization can be updated.

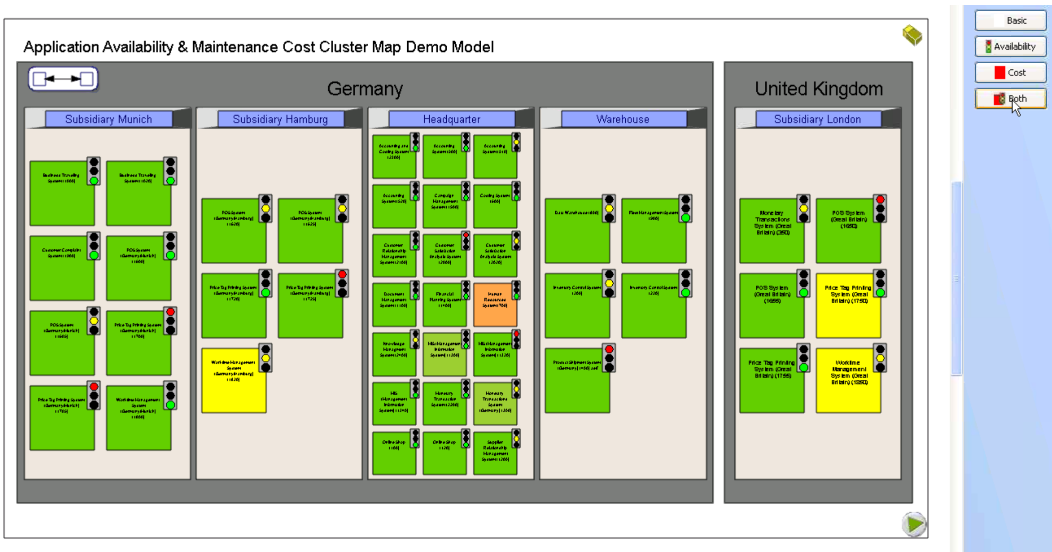


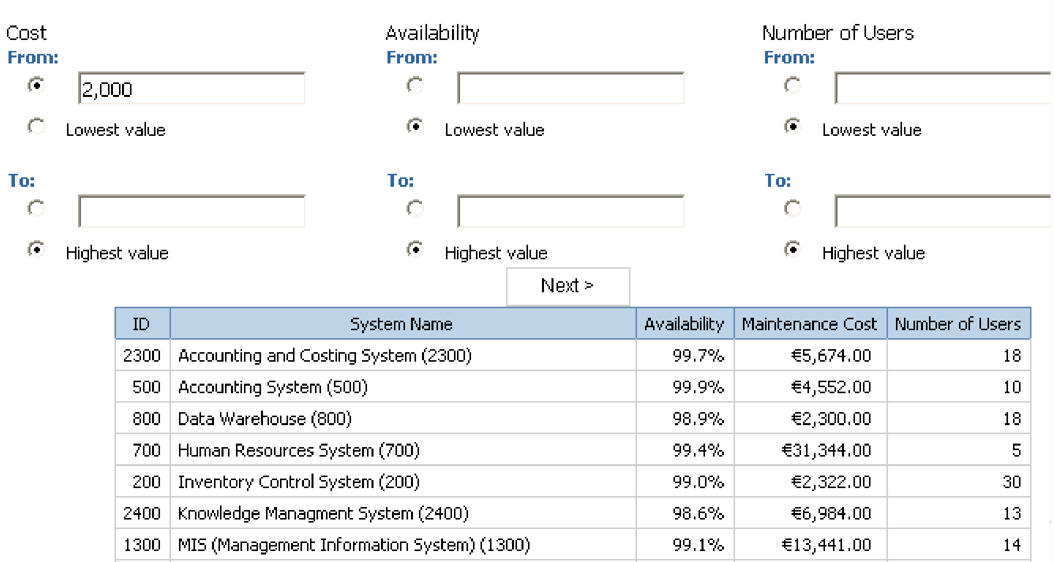
Figure 16.11.: *Troux* - axis Interacting with, Editing of, and Annotating Visualizations: Availability and maintenance costs of application systems

The cluster map shown in Figure 16.11 is an example of additional functionality, provided by *Troux*. Four action buttons realize this cluster map. The button *Basic* provides the general layout and creates the application system visualization for each location. With the button *Availability* the background coloring (green, yellow, and red) can be added showing the availability of an application system based on certain threshold values. The button *maintenance cost* adds traffic lights. Finally, the button *both* shows availability and maintenance costs together.

Troux Explorer provides several standard reports. Furthermore, new reports can be created with *Troux Intelligence*. Reports can also be annotated with certain aspects, like shown in Figure 16.12. Reports are created based on *Troux Intelligence* and are provided through the *Troux Explorer*. Figure 16.12 shows application systems and their attributes cost, availability and number of users. For all three attributes, thresholds or margins can be entered by the user to show a set of application systems based on those criteria.

Rating: 5

[4] Containers are useful for structuring data and can be opened and closed; Containers are preserved in the visualization but not committed to the repository.



ID	System Name	Availability	Maintenance Cost	Number of Users
2300	Accounting and Costing System (2300)	99.7%	€5,674.00	18
500	Accounting System (500)	99.9%	€4,552.00	10
800	Data Warehouse (800)	98.9%	€2,300.00	18
700	Human Resources System (700)	99.4%	€31,344.00	5
200	Inventory Control System (200)	99.0%	€2,322.00	30
2400	Knowledge Managment System (2400)	98.6%	€6,984.00	13
1300	MIS (Management Information System) (1300)	99.1%	€13,441.00	14

Figure 16.12.: *Troux* - axis Interacting with, Editing of, and Annotating Visualizations: *SoCaStore Troux Explorer*

16.1.4. Communication and Collaboration Support

The *Troux Explorer* allows up-to-date lightweight access to data stored within *Troux Metaverse*. The data can be viewed and edited by users, based on their roles.

Furthermore, the *Troux Explorer* has access to tabular reports from the reporting tool *Troux Intelligence*. Figure 16.13 shows the *Troux* Reporting Portal where the different reports can be accessed. Reports can be updated on demand or on a scheduled basis. Furthermore, they can be searched by the user and, if developed, reports can also be filtered like shown in Figure 16.13.

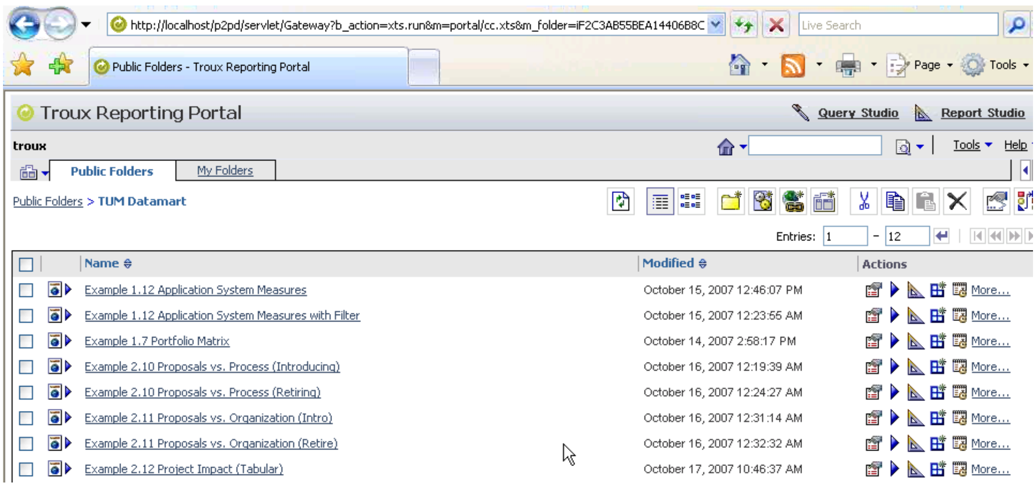


Figure 16.13.: *Troux* - axis Communication and Collaboration Support: *SoCaStore Troux Explorer*

Besides, there exists web access to visualizations. *Troux Architect* provides an HTML export for models. Visualizations can be transformed into .htm files which can be viewed through a browser. Those visualizations are static .htm files which cannot be filtered or edited.

Furthermore, the *Troux Explorer* provides an administration interface, e. g. for importing data to *Troux Metaverse*.

The *Troux Explorer* supports multiple users. It also supports a rights management which allows a role or user based access for named users with permissions attached to their ID or the groups (roles), they belong to. Restricted rights can include for example reports, models, or administrative tasks. If allowed, the rights management also provides the ability to pass given rights from users to other users. The administration of the rights management is also web-based, like shown in Figure 16.14.

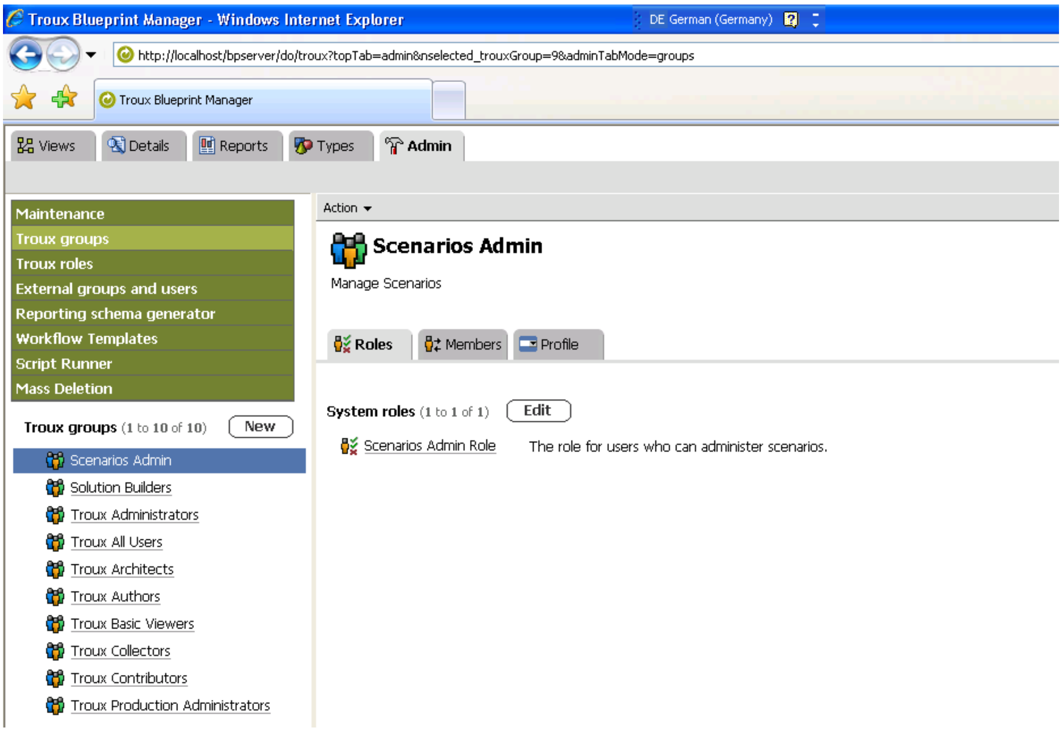


Figure 16.14.: *Troux* - axis Communication and Collaboration Support: *SoCaStore Troux Explorer*

The *Troux Architect* does not directly support multiple users. Nevertheless, as *Troux Architect* visualizations are stored on the file system they can be shared via file sharing. Thereby, it is possible to break regions of a model off into separate model files, which can then be edited by different users. However, file sharing allows only one person to work on a file at a time. If more collaborative features are demanded, visualizations can be shared using *Troux Teamserver*.

The *Troux Teamserver*, which was not tested within this survey, facilitates the collaborative work of multiple users. It is a repository solution, which stores data references and visualizations, and supports collaborative work by providing features like manage access to files, check in/check out mechanisms, timestamps, and locking of visualizations.

Rating: 3

16.1.5. Flexibility of the Information Model

The *Troux 7* provides an out of the box information model called *Troux Semantics* that has about 200 object types (classes) and about 500 relationship types. Object type components are *named type*, *properties*, *type views*, and *associated symbols* (open and closed). Relationship type components are *named type*, *properties*, *type views*, *associated symbols*, and *connect rules*.

Because it can be time consuming to create an information model from scratch, new information models can be created based on the out of the box information model or parts of it. Thereby, existing object types and relationship types can be adapted, or new one can be created, like shown in Figure 16.15. Therefore, the *SoCaStore* information model could be completely mapped and created. Furthermore, *Troux Semantics* can be adapted to allow to support frameworks such as Zachman, TOGAF, DODAF, FEAF or own corporate frameworks.

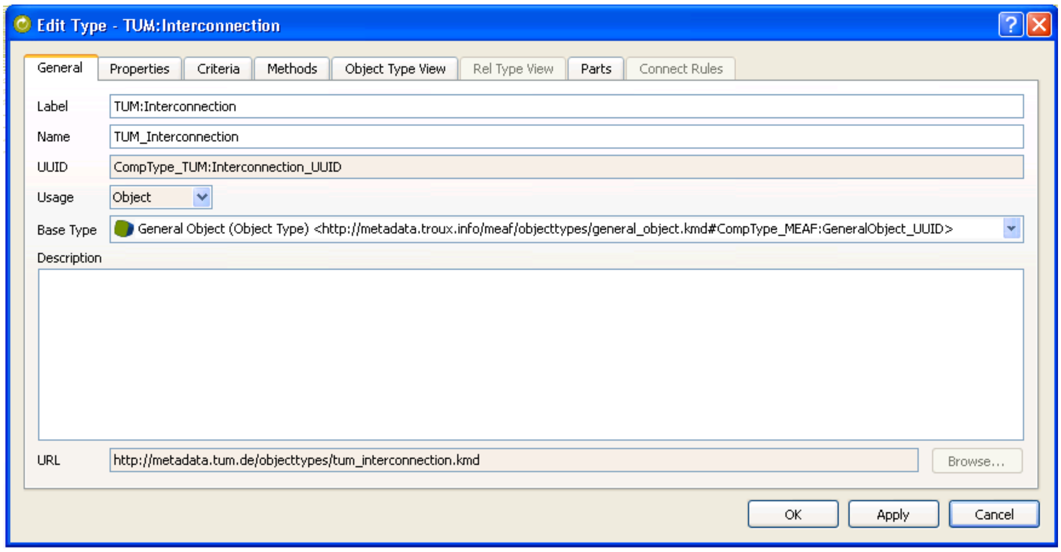


Figure 16.15.: *Troux* - axis Flexibility of the Information Model: *SoCaStore* Creation of a new Object

New attributes can be created, while predefined attributes can be adapted, hidden, or deleted. Mandatory and not mandatory values, as well as typed attributes are supported. Figure 16.16 shows the objects of the *SoCaStore* information model without relationships (for easier reading) visualized with the *Troux Architect*.

Instantiated components have both a set of property values and a set of one or more associated visualizations. The information model is based underneath a visualization, allowing only the visualization of objects and relationships which are provided by the information model.

Rating: 5

16.1.6. Support of large scale Data

This scenario evaluates how the *Troux 7* is handling application landscapes containing thousands of application systems and a respective number of interconnections. Thereby, 1000, 5000, and 10.0000 application systems with about 70 attributes, and 10 to 15 relationships to other application systems are imported into *Troux Metaverse* via the *Troux Explorer*.

As per *Troux* the *Troux 7* is architected to support the import of millions of objects and relationships. Within the *SoCaStore* environment the thousand application systems with their interconnections from the Microsoft Excel files can be completely imported. During the import an ensuring data quality process checks, if the data conforms with the given information model. After importing the data to *Troux Metaverse* the data is accessible through *Troux Architect* and the *Troux Explorer*.

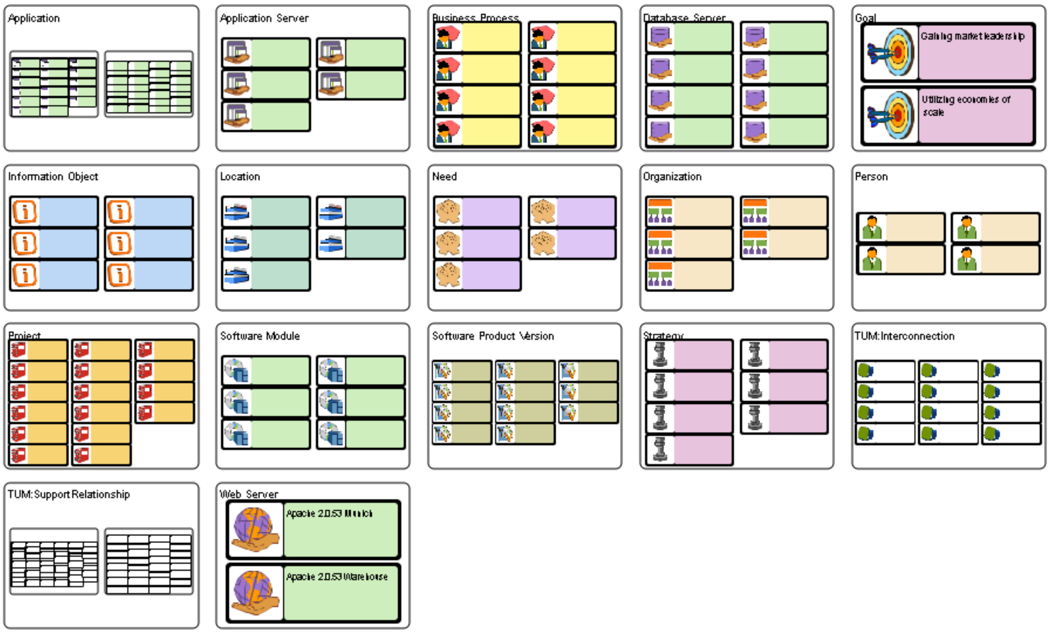


Figure 16.16.: *Troux*- axis Flexibility of the Information Model: *SoCaStore* Information Model

Within the *Troux Explorer* the access of the large scale data was considerably fast. Additionally, the creation of a report showing all application systems with their attributes, was not visibly affected by the amount of data contained.

Within the *Troux Architect* accessing the data is considerably fast using search options. Additionally, a visualization could be created containing all application systems. However, the creation took considerably long due to the manual creation process. Once created, the *Troux Architect* offers functionality for handling large scale application landscapes, like the usage of containers, which can be opened and closed, sub-modeling[5], or bundling of relationships. Using these features improves the access time considerably.

Also, editing of the application systems can be realized conveniently, because *property lists* allow the user to select several application systems and show their attributes at once. However, every application system has to be edited individually, a functionality like *changing all status to current* is missing.

Rating: 6

16.1.7. Impact Analysis and Reporting

The *Troux 7* provides several ways for supporting impact analysis. *Troux Architect* is the tool for creating and handling visualizations. Impact analysis can be realized by visualizations which can be created manually. Related objects can be dragged to a diagram and existing relationships can be displayed. Furthermore, once a visualization is created, the *browse mode* allows users to visualize certain impacts, e. g. highlighting dependencies and connections. Figure 16.18 shows an example of a swimlane diagram showing allowed options, like *Find Strategic Value Chain*, *Find Neighbors*, *Find Parts*, or *Find Threads*.

[5]Sub-modeling allows to re-use a once created visualization or parts of it.

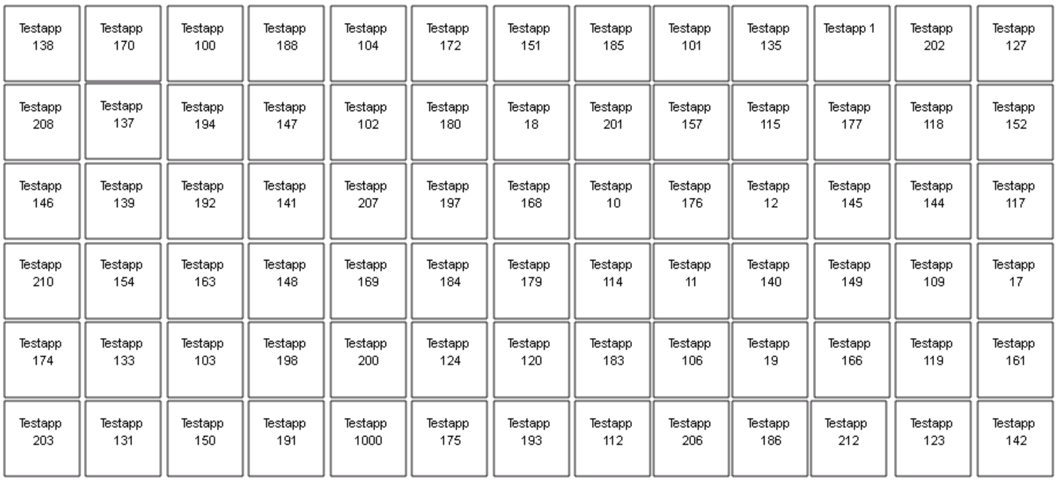


Figure 16.17.: *Troux* - axis Support of large scale Data: *SoCaStore* Large scale application landscape (cutout)

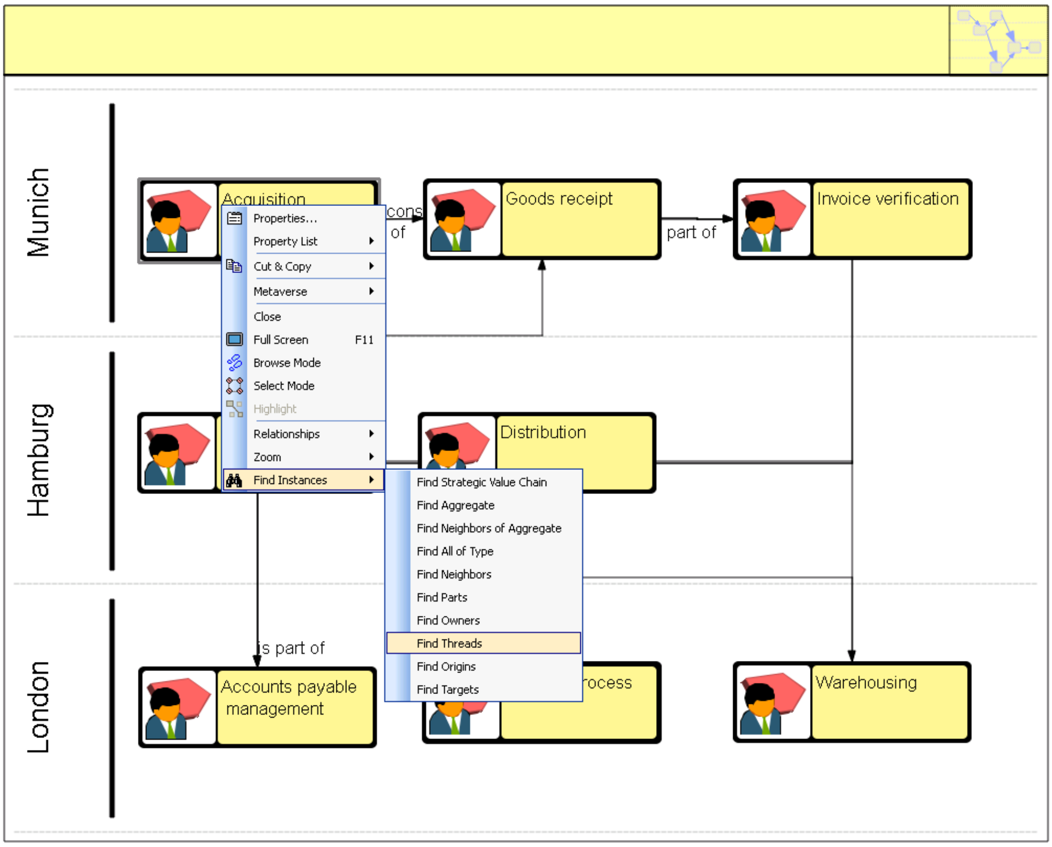


Figure 16.18.: *Troux* - axis Impact Analysis and Reporting: *Troux Architect* Swimlane Diagram with possible *Find Instances* Options

Another way for performing impact analysis can be provided by *Troux Explorer*. The user is able to access updated reports and diagrams, if the right predefined queries are provided. Figure 16.19 shows an example of an portfolio matrix provided by *Troux* consultants. It shows a portfolio matrix containing projects that are ordered regarding their *Strategic Impact Rate* and their *Return on Investment*.

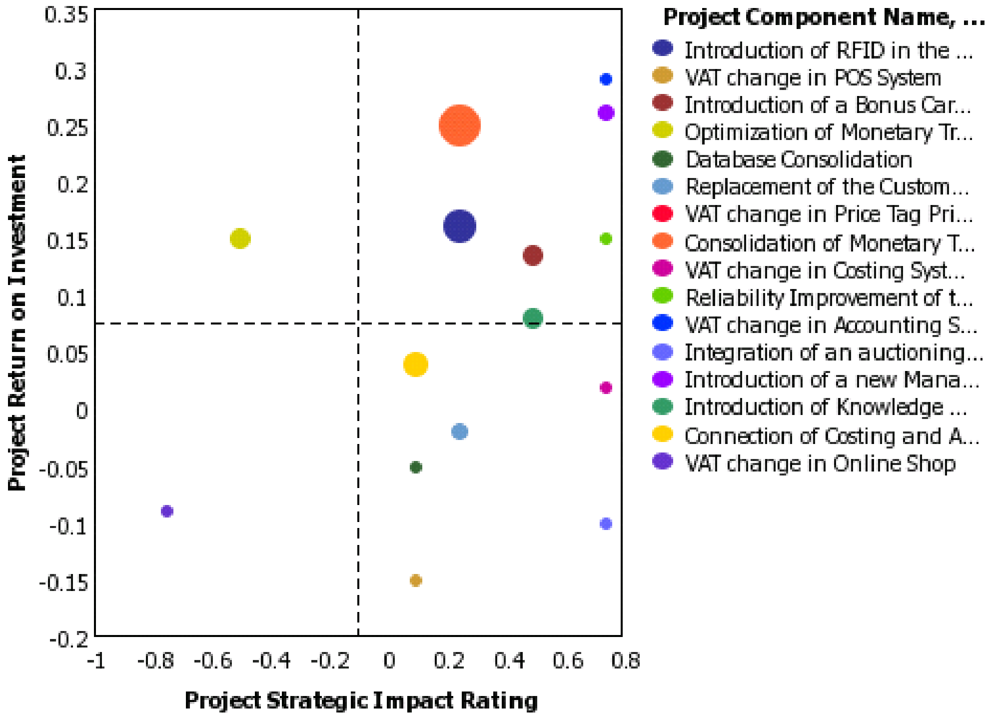


Figure 16.19.: *Troux* - axis Impact Analysis and Reporting: Example for an Impact Analysis in the *Troux Explorer*

Rating: 6

16.1.8. Usability

The *Troux 7* provides different tools with different purposes and strengths as shown in figure 16.1. *Troux Metaverse* contains the centralized information model and the centralized objects with their attributes and relationships, from where the up-to-date data can be accessed conveniently by the *Troux Explorer* and *Troux Architect*.

The web-based *Troux Explorer* contains the administration, like configuring access rights, as well as the import functionality. The import functionality is enhanced, representing status of the import and quality of the data regarding the given information model. Furthermore, the *Troux Explorer* has full access to the data within *Troux Metaverse*, which can be adapted by the customer. The *Troux Explorer* can be extended through programmable portals, and flexible solutions, like reports that can be provided based on customer requirements. Although, the *Troux Explorer* and the *Troux Architect* are not integrated, the *Troux Explorer* can provide visualizations, as they both have a connection to *Troux Metaverse*[6]. If a visualization should be provided in the *Troux Explorer* it has to be created through exporting the visualization as *.htm* file from the *Troux Architect* and put into the *Troux Explorer*, showing a certain status of the dataset.

[6]According to *Troux*, *Troux 7.1* supports the generation and presentation of certain types of visualizations, e.g. cluster maps, within *Troux Explorer*.

The *Troux Architect* is the client for creating visualizations and changing data. It is a good structured client with different windows, which can be reached by tabs. It offers a well structured menu, helpful action icons, right mouse-click functions that are context sensitive, and many helpful short-keys. Therefore, the tool handling is intuitive as needed functions are easily accessible. After a period of vocational adjustment visualizations can be created very fast. Furthermore, the tool provides a comprehensive help functionality with the sections contents, index, search, and favorites, as well as a glossary.

The *Troux Architect* is a good client for creating visualizations manually. The user has several ways for creating the visualization. On the one hand, the user can create the visualization first and store the created objects, relationships, and attributes to *Troux Metaverse*, on the other hand the user can create visualizations based on the objects, relationships, and attributes provided by *Troux Metaverse*.

While the changes of the data contained in visualizations can be propagated to *Troux Metaverse*, the visualizations of the *Troux Architect* are stored file-based and can only be passed on by file-sharing or sending the visualizations per email. The *Troux Architect* itself provides no collaboration features, like check-in / check-out, and versioning. However, if such features are needed, the *Troux Teamserver* can be installed, which provides more sophisticated multiuser support functionalities, e. g. file versioning and check in/out capabilities.

The *Troux Architect* has many features and therefore is a quite complex tool. Users need practice and experience, but then the handling of the tool can be performed quite fast. Summarizingly, *Troux Architect* provides many features to show certain aspects of the EA and perform impact analyses. However, if further functionalities, e. g. the automated creation of visualizations is needed, a certain programming effort in *VBScript* or *JavaScript* by a skilled user is necessary.

Rating: 5

16.2. Evaluation of EA Management Support

16.2.1. Landscape Management

The *Troux 7* provides a comprehensive approach to maintain and visualize the current application landscape called *As-Is* state. For the modeling of future states or scenarios of the application landscape the *Troux 7* provides a functionality called *initiatives*. *Initiatives* are provided through the *Troux Explorer*.

An *initiative* can be seen as a portfolio of objects, like application systems, that can be grouped together and represent the application landscape or parts of it at certain points of time usually in the future. Each initiative can be broken down into scenarios (e. g. one scenario contains all application systems at a certain organizational unit). Furthermore, each initiative can show as much future states at certain points of time as needed. Within each initiative, the future state can be freely modeled, like setting an application system to the status *retirement* without affecting the original *current* application system, or adding a new one. For a fast creation of a new state, it is also possible to *clone* another state and make adoptions to it. Therefore, this functionality provides the user a lot of flexibility. Figure 16.20 provides an example of an *initiative* showing the planned evolution of the current application landscape.

Once an initiative is created, it can be accessed via the *Troux Explorer* and the *Troux Architect*. For example, within *Troux Architect* a certain *initiative* can be selected. Having performed this, all objects of this initiative are provided for creating visualizations as needed, e.g. showing all application systems of a certain state, or showing a certain component and showing what it would look like in another state.

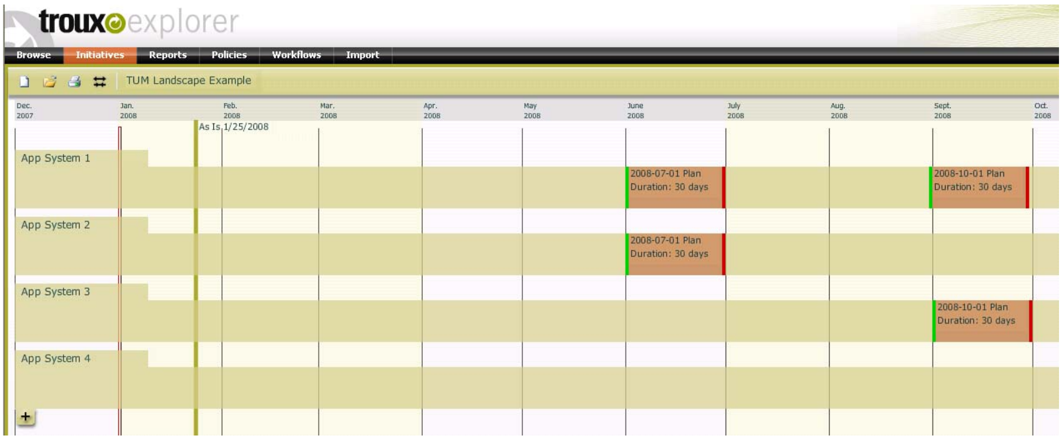


Figure 16.20.: *Troux* - axis Landscape Management: *Troux Explorer*: Example of an Initiative

Troux 7 additionally provides the so-called *reflection management* to support scalability for future state planning. As changes in the current landscape occur, those changes are automatically reflected forward throughout the states in each scenario. Thus, the user is relieved from the data management burden and the different scenarios are automatically kept up to date.

With the *initiative functionality* an overview can be created showing an initiative with its different scenarios and states. In order to transform one or more objects from an initiative directly to the current application landscape, the *state populator* or the *Troux Architect* can be used. An automatic history, that shows the application landscape at a certain point of time in the past, is not maintained. However, once a visualization is created and accessed via file share, it can be duplicated to create a history manually. Furthermore, within an initiative a comparison report can be created, showing the differences between objects and their values to another state. The difference report can be exported to Microsoft Project as Microsoft Project xml. Thereby, a tag for every difference is created that can be read by Microsoft Project as project task.

The procedure consistency is mostly given, as the created initiatives can be accessed via *Troux Architect*and *Troux Explorer*. However, a *SupportRelationship* for considering the evolution of the process support is not supported out-of-the-box.

The procedure integration is mostly given. Although the information of projects and their affects to the application landscape is stored within *Troux Metaverse*, it cannot be reused to create different future states of the application landscape.

Rating: 6

16.2.2. Demand Management

The demand management process is concerned with gathering and documenting the different demands originating from business and IT. Thereby, one or more demands may result in one or more IT projects.

The out of the box information model called *Troux Semantics* provides a class *need*, that is similar to the *demand* of the *SoCaStore* information model. Therefore, all demands can be directly imported and can be used for reports or visualizations. Once the data is imported, created, or edited, questions like: What demands exist, what attributes do they have, and what relationship to other classes do they have, can be answered.

A report listing demands and their affected application systems (see Figure 16.21) can be created manually with the web-based *Troux Intelligence* provided through the *Troux Explorer* by choosing a certain style (here *list*) and selecting the appropriate columns). The report can be easily customized, e. g. regarding the listed attributes.

Need Component Name	Need Urgency	Application Component Name	Person Component Name
Improve Customer retention	high-middle	POS System (Germany/Munich) (1600)	Mr. Maier
Improve Customer retention	high-middle	POS System (Germany/Munich) (1605)	Mr. Maier
Improve Customer retention	high-middle	POS System (Great Britain) (1650)	Mr. Maier
Adapt to government regulations	high	Price Tag Printing System (Great Britain) (1755)	Mr. Schmidt
Adapt to government regulations	high	Accounting System (520)	Mr. Schmidt
Homogenization	middle	Costing System (600)	Mrs. Huber
Improve Customer retention	high-middle	Online Shop (120)	Mr. Maier
Improve enterprise-wide knowledge management	low-middle	Customer Satisfaction Analysis System (2020)	Mrs. Huber
Reduce costs	high	Inventory Control System (200)	Mr. Hofer
Adapt to government regulations	high	Price Tag Printing System (Germany/Hamburg) (1725)	Mr. Schmidt
Adapt to government regulations	high	Price Tag Printing System (Great Britain) (1750)	Mr. Schmidt
Adapt to government regulations	high	Accounting System (510)	Mr. Schmidt
Reduce costs	high	Monetary Transactions System (Germany) (300)	Mr. Hofer
Adapt to government regulations	high	Price Tag Printing System (Germany/Hamburg) (1720)	Mr. Schmidt
Improve enterprise-wide knowledge management	low-middle	MIS (Management Information System) (1340)	Mrs. Huber

Figure 16.21.: *Troux* - axis Demand Management: Report showing gathered demands and affected application systems in *Troux Explorer*

A graphlayout map showing the direct relationships between demands and projects can be created like shown in Figure 16.22. Such visualizations can be manually created using the *Troux Architect*. Also a subset of the demands can be selected to create such visualizations, or certain demands and their relationships to projects can be highlighted. Furthermore, one project and the addressed demands can be highlighted.

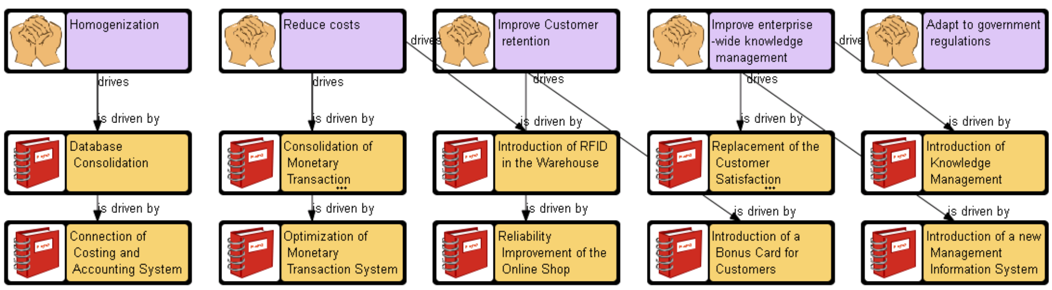


Figure 16.22.: *Troux* - axis Demand Management: Graphlayout map showing the connection between demands and projects

The procedure consistency is mostly given, as demands and their relationship to other classes, like projects and application systems can be visualized using the customization features of *Troux*.

The procedure integration is mostly given, as data previously entered can be reused in the simulation of this scenario. Nevertheless, no direct transformation from demands to projects is supported.

Rating: 5

16.2.3. Project Portfolio Management

The information model *Troux Semantics* provides classes like *project* or *need* to support this scenario. Therefore, visualizations and reports can be built, showing the different projects' influences on the application landscape. For example Figure 16.23 shows an automatically created report, providing information about how many projects affect an application system. Furthermore, it visualizes the criticality and number of accounts of the application systems to show further possible conflicts.

Number of Projects	Application	Number of Accounts	Criticality
3	Online Shop (120)	50	1
1	Accounting System (520)	10	2
1	Inventory Control System (220)	30	2
2	POS System (Great Britain) (1655)	6	1
3	Monetary Transaction System (2200)	20	1
2	POS System (Germany/Munich) (1605)	5	1
2	POS System (Germany/Hamburg) (1625)	4	1

Figure 16.23.: *Troux* - axis Project Portfolio Management: Report showing, which application systems with what criticality will be affected by how many projects

Additionally, the relationship between project proposals, or projects and thereby affected elements of the EA, like business process or organizational units can be visualized. Figure 16.24 shows a matrix that can be created through the *Troux Intelligence* providing information about which project proposal affects which business process, while Figure 16.25 shows a matrix providing information about which project proposal affects which organizational unit.

InterCalc		Distribution	Support Process	Purchasing	Accounts payable management	Invoice verification	Warehousing	Goods receipt
1.0	Integration of an auctioning platform into the online shop	x						
10.0	VAT change in POS System	x						
11.0	VAT change in Price Tag Printing System						x	
12.0	Connection of Costing and Accounting System		x					
13.0	Reliability Improvement of the Online Shop	x						
14.0	Introduction of RFID in the Warehouse	x		x			x	x
15.0	Replacement of the Customer Satisfaction Analysis System by the Data Warehouse		x					
16.0	Introduction of a Bonus Card for Customers	x						
17.0	Optimization of Monetary Transaction System	x			x	x		
18.0	Introduction of a new Management Information System		x					
5.0	Database Consolidation	x			x	x		
6.0	Consolidation of Monetary Transaction Systems	x			x	x		
7.0	VAT change in Accounting System		x					
8.0	VAT change in Costing System		x					
9.0	VAT change in Online Shop	x						

Figure 16.24.: *Troux* - axis Project Portfolio Management: Report showing, which project proposal affects which business process

Impact analyses for documenting measures like project costs or economic impact can also be created by customization, based on the customers needs. Figure 16.19 shows an example of a customized portfolio matrix, which can either be created by a skilled user utilizing the customization features of *Troux* or by *Troux* consultants. It shows a portfolio matrix containing projects that are visualized regarding their estimated *Strategic Impact Rate* and their expected *Return on Investment*.

Furthermore, the *Troux 7* provides interfaces through *Troux Collection* and *Troux Composer* to import and export objects to other Project Management tools, like Microsoft Project, that provide further solutions regarding the support of projects. Those interfaces can be enhanced by the *Troux Team* based on customer needs.

The procedure consistency is mostly given, as objectives can be achieved. However, manual and customization effort is required.

InterCalc		Subsidiary Munich	Headquarter	Subsidiary Hamburg	Warehouse	Subsidiary London
1.0	Integration of an auctioning platform into the online shop		X			
10.0	VAT change in POS System	X		X		X
11.0	VAT change in Price Tag Printing System	X		X		X
12.0	Connection of Costing and Accounting System		X			
13.0	Reliability Improvement of the Online Shop		X			
14.0	Introduction of RFID in the Warehouse				X	
15.0	Replacement of the Customer Satisfaction Analysis System by the Data Warehouse		X			
16.0	Introduction of a Bonus Card for Customers	X		X		X
17.0	Optimization of Monetary Transaction System	X	X	X	X	X
18.0	Introduction of a new Management Information System		X			
5.0	Database Consolidation	X	X	X	X	X
6.0	Consolidation of Monetary Transaction Systems	X	X	X	X	X
7.0	VAT change in Accounting System		X			
8.0	VAT change in Costing System		X			
9.0	VAT change in Online Shop		X			

Figure 16.25.: *Troux* - axis Project Portfolio Management: Report showing, which project proposal affects which organizational unit

The procedure integration is mostly given, as data previously entered into the tool can be reused, although in the application landscape management created *initiatives* cannot be directly used as input for new projects.

Rating: 5

16.2.4. Synchronization Management

The *Troux 7* provides synchronization management on a basic level. *Time interval maps* are not supported, however there exist other ways like *reports*, *impact analysis*, or a function called *reflection* that provide some synchronization management functionalities.

Figure 16.26 shows a customized report listing projects with their affected application systems and other projects that run on those application systems occurring possible conflicts. Furthermore impact analyses can be created, for example inquiring the projects affected by the delay of another project through their interdependencies on application systems or organizational units.

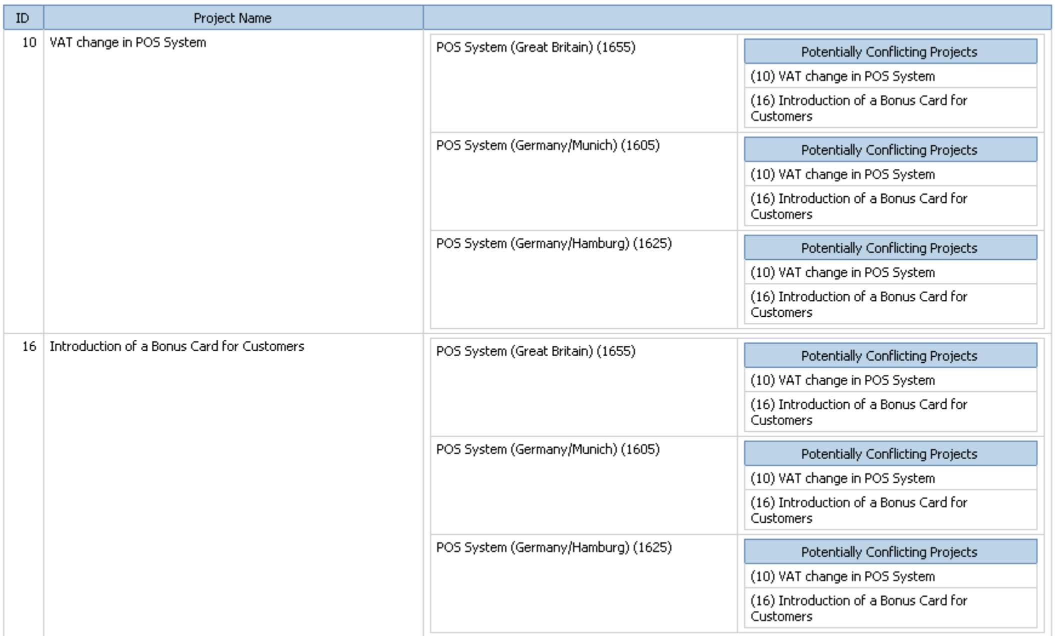
Reflections are reports that notify the user, when a component, either an object or a relationship, changes prior to planned future changes. One example would be, if, a user starts to use an application system in a visualization, e. g. creates a new relationship to this application system, although the application system should be retired a month later. With the *reflection report*, the user would be notified about this conflict. The *reflection report* can be sent by email or can be accessed from the initiative tab. However, the notification is not directly triggered, when the observed object is adapted, but when the reflection report is run.

If more synchronization management functionality is needed, the tool provides interfaces through *Troux Collection* and *Troux Composer* to realize a connection between the *Troux 7* and Project Management tools, like Microsoft Project. The typical synchronization management tasks like exploring dependencies and interconnections between projects can be then performed by these tools. The interfaces can further be enhanced by the *Troux Team* based on customer needs.

The procedure consistency is partially given, as e. g. a time interval map cannot directly be created.

The procedure integration is mostly given, as data from the project management can be reused. Nevertheless, automatic impact analysis concerning projects affected from the delay are not directly supported.

Rating: 3



ID	Project Name		
10	VAT change in POS System	POS System (Great Britain) (1655)	Potentially Conflicting Projects (10) VAT change in POS System (16) Introduction of a Bonus Card for Customers
		POS System (Germany/Munich) (1605)	Potentially Conflicting Projects (10) VAT change in POS System (16) Introduction of a Bonus Card for Customers
		POS System (Germany/Hamburg) (1625)	Potentially Conflicting Projects (10) VAT change in POS System (16) Introduction of a Bonus Card for Customers
16	Introduction of a Bonus Card for Customers	POS System (Great Britain) (1655)	Potentially Conflicting Projects (10) VAT change in POS System (16) Introduction of a Bonus Card for Customers
		POS System (Germany/Munich) (1605)	Potentially Conflicting Projects (10) VAT change in POS System (16) Introduction of a Bonus Card for Customers
		POS System (Germany/Hamburg) (1625)	Potentially Conflicting Projects (10) VAT change in POS System (16) Introduction of a Bonus Card for Customers

Figure 16.26.: *Troux* - axis Synchronization Management: Report showing possible conflicts of project proposals

16.2.5. Strategies and Goals Management

Strategies and goals management addresses issues of aligning the EA management activities to the organization's strategies and goals. Thereby, tasks like decomposing strategies into smaller and more detailed pieces, and addressing them by specific action items, have to be supported.

The *Troux 7* addresses such requirements. The *Troux Semantics* provides concepts for strategy and goal classes out of the box. Attributes, like *reached*, *urgency*, or *strategic value* are provided, or can be added, as well as relationships between classes are provided or can be added. Therefore, questions like, which strategy leads to which goals, can be addressed by visualizations like Figure 16.27, or using a relationship matrix like the one shown in Figure 16.28, can be created utilizing *Troux Intelligence*. Further reports can be created with *Troux Intelligence*, answering questions regarding the strategies and goals management like, the fulfillment of the individual goals.

The procedure consistency is mostly given, if the information model is adapted to support needed relationships, e. g. between goals and projects and the customization features of the tool are utilized.

The procedure integration is completely given. Information previously entered into *Troux 7* can be reused in this context.

Rating: 5

16.2.6. Business Object Management

The Business Objects Management can be executed using the *Troux 7* similar to e. g. the *Strategies and Goals Management* or the *Demand Management*. The *Troux Semantics* provides classes and relationships that support business object management, like *business object* and *business process*. However, to support all requests of the SoCaStore, additional attributes have to be added.

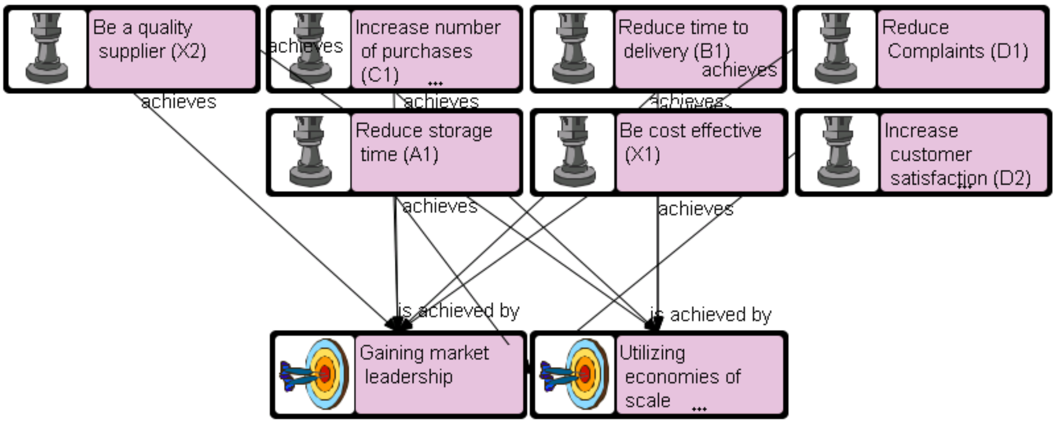


Figure 16.27.: *Troux* - axis Strategies and Goals Management: Graphlayout map connection between strategies and goals

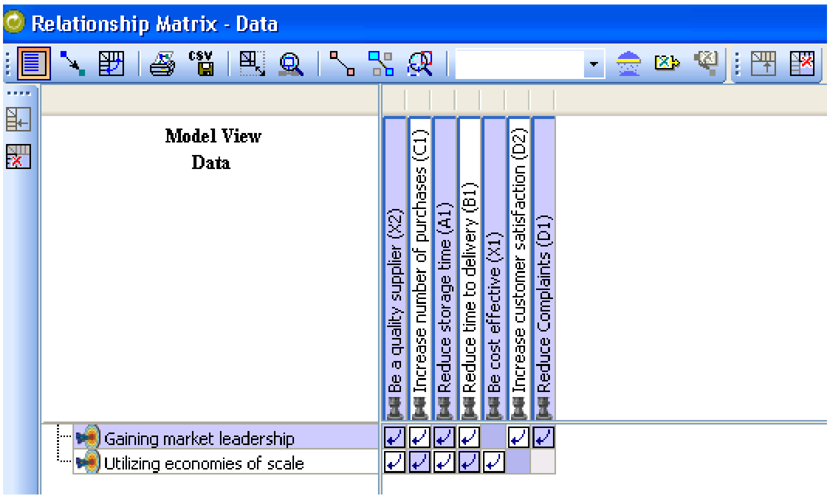


Figure 16.28.: *Troux* - axis Strategies and Goals Management: Relationship matrix between strategies and goals

In the *SoCaStore* simulation, the most important class concept for this scenario *business object*, *business process*, and *application systems* are connected over a support relationship, which had to be added to the information model.

For creating the required visualizations or reports, the customization features of the tool can be utilized. The traversing over the support relationship to get the required information, as well as the hiding of certain objects in between, is supported by the *Troux 7*. However, it requires manual effort and deeper knowledge of the user in creating queries that support traversing.

The requirements of this scenario include data flow between the application systems performing operations on the business objects and the kind of operations performed in a specific application system. The *SoCaStore* information model provides the four operations *create*, *read*, *update*, and *delete* that can be performed on business operations. Figure 16.29 shows a report providing business objects and the operations performed on them by application systems[7].

[7]The report was provided by *Troux* consultants utilizing the customization features of the tool.

		Customer Complaint				Customer				Monetary Transaction				Invoice				Stock Item				Price Tag			
		C	R	U	D	C	R	U	D	C	R	U	D	C	R	U	D	C	R	U	D	C	R	U	D
Subsidiary London	Customer Complaint System (1900)	×		×			×																		
	Inventory Control System (200)													×	×				×	×	×				
	POS System (Great Britain) (1650)						×			×									×	×			×		
	POS System (Great Britain) (1655)						×			×									×	×			×		
	Monetary Transactions System (Great Britain) (350)									×	×														
	Campaign Management System (1500)						×	×											×						
Subsidiary Munich	Customer Complaint System (1900)	×		×			×																		
	POS System (Germany/Munich) (1600)						×			×									×	×			×		
	Inventory Control System (200)													×	×				×	×	×				
	POS System (Germany/Munich) (1605)						×			×									×	×			×		
	Monetary Transactions System (Germany) (300)									×	×														
	Campaign Management System (1500)						×												×						
Subsidiary Hamburg	Inventory Control System (200)													×	×				×	×	×				
	POS System (Germany/Hamburg) (1625)						×			×									×	×			×		
	POS System (Germany/Hamburg) (1620)						×			×									×	×			×		
	Monetary Transactions System (Germany) (300)									×	×														
	Campaign Management System (1500)						×	×											×						
	Customer Complaint System (1900)	×		×			×																		
Warehouse	Inventory Control System (200)													×	×				×	×	×				
	Product Shipment System (Germany) (400) Jeff									×								×	×	×					

Figure 16.29.: *Troux* - axis Business Object Management: Report showing business objects and the operations performed on them by application systems within *Troux Explorer*

The procedure consistency is mostly given, as customization effort is required regarding the adaption of the information model and the creation of the visualizations and reports.

The procedure integration is completely given, as data previously entered in the tool can be reused in the context of this scenario.

Rating: 5

16.2.7. SOA Transformation

This scenario consists of two parts. On the one hand, it requires profound knowledge of the current application landscape to derive possible SOA transformations, on the other hand, it requires the ability to model the performed changes. As described in the section *application landscape management* the *Troux 7* provides a comprehensive approach to visualize and to create reports regarding the current application landscape, called *As-Is* state.

Therefore, questions like, which functionalities are applicable candidates for the transformation into a business service, or which application systems are used by numerous users, can be answered by creating the appropriate visualizations or reports. For example, Figure 16.11 shows a cluster map of the application systems with their availability and maintenance costs. Figure 16.23, shows a report, which lists the application systems regarding their criticality, and number of users working with the system. Figure 16.29 shows a customized report, which can either be created by a skilled user utilizing the customization features of *Troux* or by *Troux* consultants. The report provides information about business objects and the operations performed on them by application systems. Thus, potential SOA transformation candidates can be derived.

Once applicable SOA transformation candidates are found, the *Troux initiatives* can be used to model future states of the application landscape (Section 16.2.1). Initiatives are based on the used information model. Therefore, if the information model does not reflect classes to support SOA classes and their attributes, like business services, or service level agreements, the information model has to be adapted first (For more information see Section 16.1.5). To derive changes that should be performed during the transformation to a SOA, the function of *reflection* (A detailed description can be found in Section 16.2.4) can be used.

The procedure consistency is mostly given, as customization effort is required enhancing the information model regarding SOA related classes and relationships to support this scenario and the creation of the visualizations and reports.

The procedure integration is completely given, as data previously entered in the tool can be reused in the context of this scenario.

Rating: 6

16.2.8. IT Architecture Management

The IT architecture management deals with the introduction and implementation of architectural solutions standardizing the architecture of specific application systems to address heterogeneous application landscape.

To address this scenario, new classes and their relationships, *blueprint element*, *solution element*, *architectural blueprint*, and *architectural solutions* have to be introduced to the *Troux* information model. For example, the class *architectural solution* has to be created and enhanced with attributes, like *high availability* and *modularization*. Besides, to achieve the objectives of the scenario the application systems have to be addressed, that are connected with the architectural blueprint traversing over the application systems.

A graphlayout map can be created showing architectural solutions and used technologies like the one shown in Figure 16.30. However, a cluster map, showing application systems and their compatibility with the architectural solution by using color-coding, e. g. green for conformance and red for non conformance, can only be realized utilizing an additional template that is available via *Troux*.

A report, which lists all application systems using a selected architectural solution can be created utilizing the *Troux Explorer*. Because the report requires traversing from application systems to architectural blueprints, the report requires deeper user knowledge in creating queries that support traversing or initial support by *Troux* consultants.

The procedure consistency is mostly given, but requires customization effort.

The procedure integration is completely given, as data previously entered can be reused in the simulation of this scenario.

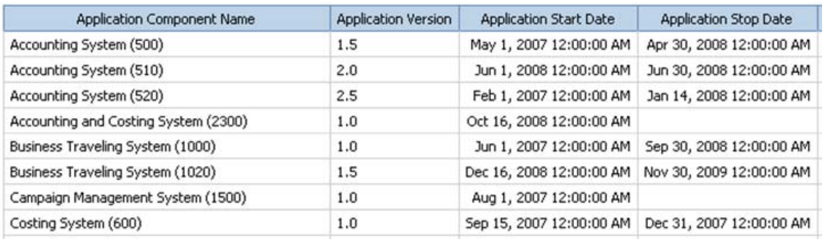
Rating: 6

16.2.9. Infrastructure Management

The infrastructure management is achieved with the *Troux 7* quite similar to the IT architecture management (see Section 16.2.8). The predefined information model *Troux Semantics* provides a lot of classes and relationships that can be used for this scenario. Furthermore, the information model can be adapted to fully represent the *SoCaStore* information model.

The *SoCaStore* intends to consolidate its database systems to decrease the costs for maintenance and licenses. To support this scenario, different reports and visualizations can be created manually. For example a report can be created, showing which application systems are in use and how long. Figure 16.30 provides such a report representing application systems with their version, their start date, and their retirement date. A similar report can be created for databases, showing their costs and running out of support. Furthermore, the data can be exported to Microsoft Excel and this tool can be utilized to realize further reports.

Visualization that do not require traversing can be created manually without adaptations. For example a graphlayout map can be created, like shown in Figure 16.31, which visualizes the connection between architectural solution and used technologies.



Application Component Name	Application Version	Application Start Date	Application Stop Date
Accounting System (500)	1.5	May 1, 2007 12:00:00 AM	Apr 30, 2008 12:00:00 AM
Accounting System (510)	2.0	Jun 1, 2008 12:00:00 AM	Jun 30, 2008 12:00:00 AM
Accounting System (520)	2.5	Feb 1, 2007 12:00:00 AM	Jan 14, 2008 12:00:00 AM
Accounting and Costing System (2300)	1.0	Oct 16, 2008 12:00:00 AM	
Business Traveling System (1000)	1.0	Jun 1, 2007 12:00:00 AM	Sep 30, 2008 12:00:00 AM
Business Traveling System (1020)	1.5	Dec 16, 2008 12:00:00 AM	Nov 30, 2009 12:00:00 AM
Campaign Management System (1500)	1.0	Aug 1, 2007 12:00:00 AM	
Costing System (600)	1.0	Sep 15, 2007 12:00:00 AM	Dec 31, 2007 12:00:00 AM

Figure 16.30.: *Troux* - axis Infrastructure Management: *Troux Explorer*: Visualization showing architectural solutions and used technologies

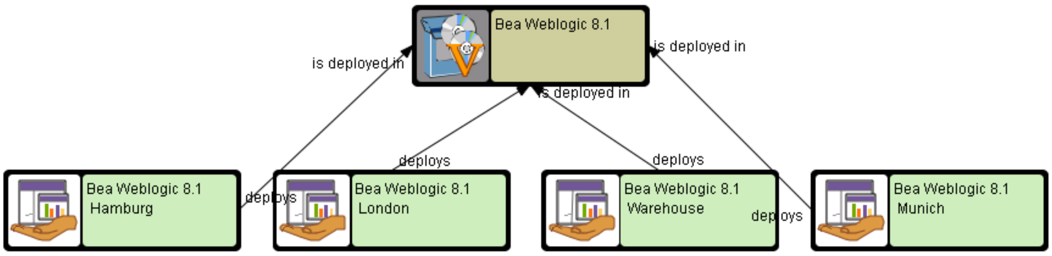


Figure 16.31.: *Troux* - axis Infrastructure Management: Graphlayout map showing architectural solutions and used technologies

Furthermore, a relationship matrix can be created, e. g. showing the relationships between application systems and their used technologies like shown in Figure 16.32. Visualizations that require traversing can also be created manually, based on the provided information model. However, therefore, a skilled user is needed, who has deeper knowledge in creating queries that support traversing.

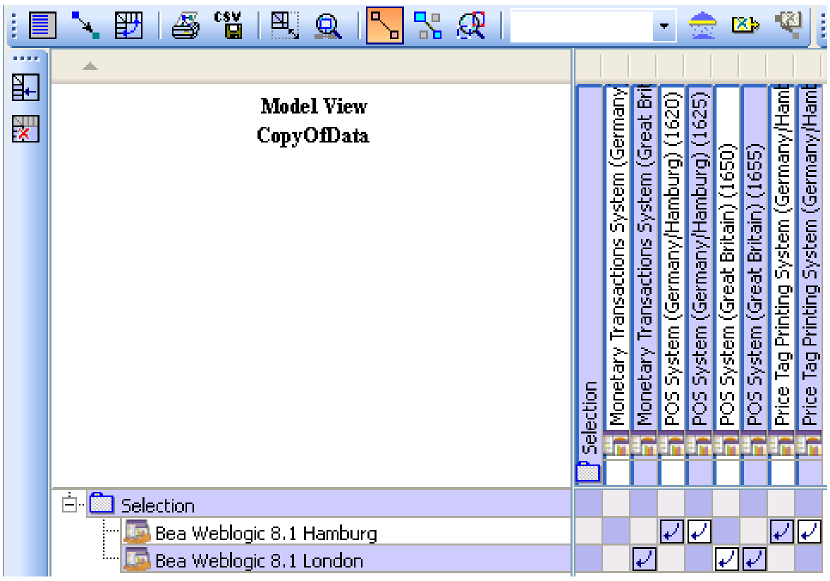


Figure 16.32.: *Troux* - axis Infrastructure Management: Relationship matrix showing application systems and their used technologies

The procedure consistency is mostly given, although manual effort and deeper knowledge of the user is required.

The procedure integration is completely given, as data previously entered can be reused in the simulation of this scenario.

Rating: 4

16.3. Tool Vendor's Profile

Matt Price, Vice President

Troux Technologies provides the largest Global 2000 enterprises and governments with strategic IT planning software that helps IT to keep pace with increased business change without adding operational risk. *Troux*'s breakthrough strategic IT planning software unlocks the full power of Enterprise Architecture to provide IT leaders with instant access to reliable intelligence, better insight into complex interdependencies and comprehensive impact analysis through scenario planning. As a result, IT teams can accelerate and scale business change initiatives, with faster and more informed decisions, expertly executed initiatives and unprecedented accountability to the business – helping organizations to reduce costs, minimize risks and enhance business agility. To learn more visit www.troux.com.

Troux Advantage

Troux is widely regarded as the predominant innovator in the field of Strategic IT Planning and Enterprise Architecture. Our work at a strategic level with very large global organisations has helped to forge a mature software platform which uniquely handles the scale and complexity that their businesses demand. Many of these companies came to *Troux* after trying and failing to get departmental planning or functional modeling tools to give them the cross-enterprise visibility and planning control they needed. As a result, *Troux* customers deliver transformations quicker, with less risk and with wider-reaching impact. The planning intelligence that *Troux* provides gives organisations the confidence to set, manage and meet more aggressive transformation targets and then proactively set and manage policy for ongoing IT excellence.

Troux Innovation

Troux was the first company to integrate the four critical components for Strategic IT Planning into a flexible and open platform architecture that connects neatly to existing enterprise tools and software. Each of these components is developed to have 'best-of-breed' capabilities: *Troux* Architect™ for sophisticated Enterprise Architecture Modeling; *Troux Metaverse* for storing enterprise planning information such as models and instance data; *Troux Collection* for gathering and maintaining information from across the enterprise and *Troux Intelligence* for sophisticated business intelligence for IT. Two recently released innovations have met with broad acclaim: *Troux Initiatives* clarify how business transformations - and their associated IT projects - will impact the bottom line of the organization. *Troux Composer* enables customer subject-matter experts to quickly assemble solutions which address specific transformation or management needs. Finally, *Troux* was the first company to release out-of-the box platform-based solutions for two of the highest priority transformation requirements: Application Portfolio Optimization and Standards Management.

Troux Results

Market momentum for *Troux* continues to build. Recent record financial results have been driven by the addition of blue chip customers such as Fiducia, Vodafone, and AstraZeneca to an existing worldwide base of more than 200 customers. The open nature of the platform has lead to the solution being widely deployed by global leading systems integrators Accenture and EDS. *Troux* is ranked as a 'Leader' in the Gartner Magic Quadrant for Enterprise Architecture Management. Excerpts from the recent Butler Group report 'The Transformation Capability of Enterprise Architecture', highlight the value that *Troux* is bringing to chip manufacturer AMD.

'An active user of Troux Technologies' EA tools, [AMD] has found the toolset to be very powerful.

... Modeling such information manually would be too time-consuming, and by the time any model was complete it would be significantly out of date, and this is where Troux's technology comes into play. The Collector component of Troux automates much of the upload of information, such as types and numbers of servers, application instance data, and various deployed entities. This provides much information to the Troux repository in near real time.

... The AMD team is currently considering extending the information that it captures about the way the company does business beyond the traditional EA area, into the wider business. The extensible nature of the toolset brings the ability to use it for a wide range of real business intelligence challenges.'

www.troux.com/resources/resource_library/whitepapers/whitepaper_butler_transformation.pdf

CHAPTER 17

Summary

Contents

Compared to the results of the Enterprise Architecture Management Tool Survey 2005, EA management presents itself in 2008 as a more mature discipline, better understood by practitioners as well as by tool vendors. Nevertheless, despite the undisputed evolution of the field, actual convergence has not been reached, neither concerning the approaches to EA management nor the market for EA management tools. In the latter area, a multitude of changes has occured since 2005, continuing the tendency, we had experienced that time – the market is on the move. Notwithstanding, the overall number of vendors seems to have not decreased at all; the key players in the area remain and show no sign of weakness.

Concerning the approaches to EA management incorporated in the different tools, the diversity indicated at the end of the past survey and again detailed below did not decrease. In contrast, it has ever since evolved to a unique selling proposition for tools far beyond an approach, which could be considered as an accidental heritage from the tool vendor's origin. In Section 17.1, we detail on these approaches forming the *dimensions of distinction*, which span the solution space of the current market for EA management tools.

The practitioners, as alluded to above, have also gained additional insight into EA management, drawn from management endeavors in this field, which have since 2005 been undertaken in the majority of the large enterprises and even in a multitude of medium-sized companies. With this increasing importance and proliferation of EA management in practice, also the requirements and demands on a tool support for EA management have developed further. These increased and changed requirements are reflected by the adapted scenarios (see Chapter 4) in the survey, which have undergone a major rework since 2005. Complementing this, a few completely new scenarios have been added, most prominent the *SOA transformation* scenario as of Section 4.2.7. These changes in requirements mirror the importance of EA management in practice as the *glue* between a plethora of different areas of management in a company. Section 17.2 summarizes our experiences with the tools in general concerning specific EA management tasks, especially targeting the new ones.

The survey of 2005 was concluded by a discussion on *aligning business and IT*. There emphasized at the seclusiveness, IT departments are likely to work in. Due to the resulting lack of communication between business and IT, it is likely that IT partially neglects business demands. From this point, we elaborated on the need for a more holistic view on both IT and business aspects, i. e. on the whole *Enterprise Architecture*, in order to foster the alignment between business and IT. In retrospection, it becomes obvious, that EA management has received significantly increased attention since that time, although the tendency, that the IT departments are the drivers of EA management endeavors, prolongs. This might also be reflected by the documentation and planning focus current EA management approaches take. Complementing this, we see a variety of new aspects and directions arising in the context of managing the EA. A final Section 17.3 gives some indications on these possible topics and sketches promising ideas, we regard to be a part of, or closely related to EA management.

17.1. Approaches of EA Management Tools

Every EA management tool analyzed in this survey is based on its individual approach to EA management. These approaches differ widely and cannot be compared in a simple one-dimensional manner – especially, as no approach has yet proven to be the *best*. However, some approach may be suited better for a specific user with a specific usage scenario. We have identified three prominent dimensions, a tool supported approach to EA management can be classified in. Each dimension is characterized by the following choices:

1. **flexibility vs. guidance**,
2. **preconfigured vs. customization**, and
3. **integration vs. single-point-of-truth**.

Flexibility vs. Guidance

The first dimension, *flexibility vs. guidance*, was introduced in the final chapter of the 2005 version of the survey[1], establishing the three different options *metamodel driven*, *methodology driven*, and *process driven*. These options actually can be used to describe the tools evaluated in this survey. Subsequently, we sketch the main properties for each option.

The *metamodel driven* approach grants the user maximum flexibility concerning the adaptation of the information model of the tool. Therefore, strong metamodeling capabilities are incorporated and reflected in the underlying repository. These capabilities may nevertheless vary from tool to tool; some tools adhere to standardized metamodeling facilities, e. g. the *Meta Object Facilitiy (MOF)*, while other tools bring their individually developed facilities, which may exert limitations to adaptability of the information model. Complementing the mechanisms for adapting the information model, the tools pursuing a metamodel driven approach commonly provide facilities for configuring and adapting the visualization techniques in accordance to the changes in the underlying information model. A tool adhering to a metamodel driven approach may be especially beneficiary for companies, which already have developed their own approach to EA management, potentially complemented by an individual information model tailored for their specific demands. Companies, which – in contrast – are on the way of introducing EA management *from scratch* may need additional guidance for developing and implementing an information model best suited for supporting their way of managing the EA.

In contrast, some tools evaluated pursue the *methodology driven* approach, commonly providing a comprehensive predefined information model together with a set of predefined visualizations, reporting and analysis techniques. In these tools, only minor adaptations to the underlying information model

[1]Nevertheless, it was not directly called *flexibility vs. guidance* there, but the approaches were presented

are possible, e. g. the introduction of new attributes. The core concepts of EA management are therefore covered by the predefined information model to variable extent. Further, the tools differ concerning the provision of a semantics for the concepts defined in the information model – ranging from the usage of descriptive names for the concepts to an all-embracing documentation of the method and its constituents but not necessarily a formal semantics. This definition of the semantics is often complemented by a description of the usage context for each of the predefined visualizations and analysis techniques. Sometimes, these techniques are realized in a configurable way, such that the user can extend them to also take newly introduced attributes, i. e. the information model adaptations, into consideration. A tool adhereing to a methodology driven approach may be useful for companies, which would like to execute EA management based on a well-founded methodology, instead of developing a fully individual approach. Nevertheless, when implementing EA management in the organization as a management process, a company has the flexibility to choose the best suited implementation, based on the organizational roles, which might have been introduced by the tool's methodology.

The *process driven* approach provides maximum guidance for EA management. This approach can be seen as an extension to the methodology driven one, complementing the predefined information model, reports, and analyzes with a defined workflow and set of user roles. Thereby, a whole EA management process is defined, providing procedures and defining activities, which have to be fulfilled in order to perform EA management. As an example of these procedures, one could think of a predefined process for creating project proposals from demands. In order to perform this, a user of the appropriate role, e. g. project manager, would have to review the demands utilizing predefined visualizations and analyses, before he decides how to group them to project proposal, which are then finally created in this activity. A tool pursuing a process driven approach may be especially interesting for companies, that are willing to adopt a best-practice process for executing EA management, e. g. as the company experiences problems with the current EA management process or has not yet established one. In adopting the process as prescribed by the tool, the company can leverage the maximum process guidance, without being burdened with the development and implementation of new-defined procedures and activities. In contrast, companies, which already have established their specific EA management process, potentially together with a tailored information model, may find a process driven tool too rigid and inflexible for supporting their individual demands.

Preconfigured vs. Customization

An actual tool, as evaluated in the survey, might not always be commited to one of these approaches exclusively. It may happen, that in different EA management tasks, different levels of guidance are available. Furthermore, a tool might out-of-the-box provide no or only basic procedure guidance, but may be customizable to implement an enterprise-specific EA management process. This customization aspect is central to the next dimension of distinction concerning EA management tools – the *preconfigured vs. customization* dichotomy.

Some of the tools, especially those pursuing a methodology or process driven approach as introduced above, are designed *EA management solutions*, which provide *preconfigured* functionalities for peforming the tasks and activities connected to the management of the EA. In order to effectively execute EA management with such a tool, consulting and training is deemed advisable to foster the organizational implementation of the procedures, which are supported by the tool. In contrast, a few tools – all pursuing a metamodel driven approach – are designed as *EA management platforms*, providing the basic capabilities necessary for realizing an organization specific EA management procedure. For these tools, consulting commonly encompasses *customization* projects, tailoring the tool to suite the individual needs of the customer. Again, the two approaches cannot be differentiated sharply, as most tools support customization to a specific extent. Nevertheless, some vendors have a strong commitment to customization and a prospective user should also consider the tailoring strategy of the respective tool, when deciding on an EA management tool.

Integration vs. Single-point-of-truth

The last dimension of distinction is again not a sharp classification, but merely indicates different characteristics concerning the tool's approach to information integration. On the one hand, a group of tools stores information about the EA in the EA management tool itself, while other information, such as project schedules or business process information should be drawn from other sources via importing mechanisms. Nevertheless, when having imported this information, it can be altered in the EA management tool in order to facilitate management endeavors targeting the EA as a whole. If updates of the information from the other management tools should be taken into account, again an import can be performed, commonly providing mechanisms for resolving conflicts. In this context, each of the tools involved in the information exchange – also including the EA management tool – is considered the single-point-of-truth for specific information, e. g. the project information. Transferring information between these tools is performed by user request via export and import mechanisms. This approach is capable of keeping data consistency high and data sovereignty issues clearly solved. Nevertheless, as EA management is commonly concerned with linking different management areas, the single-point-of-truth approach might lead to a sub-optimal information situation in the EA management endeavor. Contrasting this, some tools exist, which regard themselves as pursuing an *integration* approach. Therein, the EA management tool acts as a data warehouse for EA information, extracted, transformed, and loaded into the EA management repository via a multitude of connectors and mechanisms. The main focus of the EA management tool thereby lies on maintaining relationship information, connecting the data from different sources. This is necessary, as specific information, i. e. on the business support provided by an application system are commonly neither stored in a BPM nor in a systems management tool. Pursuing this approach, a tool is likely to provide all-embracing information about the EA, but also to rise additional complexity in keeping this information consistent and in resolving issues of data sovereignty, if information should be changed via the EA management tools interface.

In addition, to the dimensions of distinction detailed here, a multitude of other criteria for classifying the tools could be found. Furthermore, as stated above, the differentiation of the tools is often not sharp – nevertheless, keeping in mind the considerations from this chapter during the selection of an appropriate EA management tool could be beneficiary for a company.

17.2. Lessons Learned

As manifold as the approaches to EA management pursued by the tools, one would also expect the strengths and weaknesses to be. This has not been the case in 2005 and still not is, these days. Below, we sketch some common weaknesses and give some indications on their potential causes.

Having a look on the overall results of the tools as ordered on the axes, it is obvious, that there are axes, where we have decided not to rank a tool at seven. On these axes, none of the tools has been capable of directly delivering all demanded objectives. To give a prominent axis, where this has been the case, we take a closer look on the *SOA transformation*. Here we expected the tools to provide support for determining service candidates, e. g. based on information about business support. Not surprisingly, many of the tools analyzed provided a concept *service*, which could in most cases be used instead of an application system – nevertheless, the more specifics of a service and thereby of a service oriented architecture (as shorthanded in Section 4.2.7) were not directly reflected in the tools. Therefore, a user might have the option to model services instead of application systems, but having modeled the current systems first, the user is not provided any support for deciding, which of these applications should be transformed into one or more services. One might speculate, why this is the case – we see the following potential reason: A multitude of approaches on how to transform the EA into a SOA exist and are discussed, but none of these approaches has yet proven to be the most succesful one. Therefore, tool vendors might abstain from implementing a vast variety of approaches, in order to not bewilder the user. Implementing just one of these approaches might nevertheless result in an overcommitment to a SOA transformation methodology not yet being completely mature.

As a follow-up of the 2005 version of the survey, the topic of project management integration of the tools was repeatedly discussed. It is obvious, that the situation has changed since then, although still no tool was able to fully satisfy the EA management demands originating from this field, especially when *synchronization management* (cf. Section 4.2.4) is considered. While most tools have been able to govern basic information on the project schedule, as *start* or *end date*, it was not always possible to link these information to the planned application landscapes as of the corresponding date. Even with tools providing capabilities in this respect, another limitation showed up, concerning the management of *project delays*. Surprisingly, no tool evaluated had an *out-of-the-box* concept to model project delays, which may be considered an unwanted but not uncommon phenomenon in practice. Therefore, no direct support for the delayed planning states of the EA or parts thereof was provided. This would nevertheless be an aspect of undisputed importance for EA management, although one might argue, that the delay itself was a concept from project management, not to be a cornerstone of an EA management endeavor.

17.3. Possible topics in EA Management

Concluding this survey, we would like to rise some topics, which we find interesting in the context of EA management tools. The tools, as of today, bring along a plethora of functionalities for facilitating the management of the EA in a company. These functionalities are useful for documenting and planning the dense web of interdependencies, which is made up by the constituents of the EA in general and the application landscape in particular. The capabilities are further complemented by visualization mechanisms, which are helpful for getting insights on and an overview of the whole EA.

Nevertheless, when starting with a tool supported EA management endeavor, a company is likely to find the initial steps quite discomforting. The repository of the tool is likely to contain nothing but a set of exemplary data on the fictional EA of a fictional company, if any. Starting with that, a user has to get all the information needed for EA management in, regardless of the fact, that some of this information could be seen as *standard* data, a multitude of EA management tool users is likely to gather – standard data, e. g. lifecycle and support time information on operating systems.

There we see potential for improvement, which could be realized by creating a platform for exchanging such data among EA management pracitioners and software vendors. This platform could be helpful to relief a company newly introducing an EA management tool from the necessity to gather data of that kind, further providing quick access to initial and useful content. Complementing this, such a platform would rely on a common format for exchanging data between different EA management tools, which we have pointed out being a valuable contribution in this field ever since the 2005 version of the survey. In this context, standard formats for exchanging information about object oriented models, e. g. the *XML metadata interchange (XMI)* format could be considered, as such format would further foster the integration to UML tools and thereby to the software development processes. Also other formats, especially from the fields of software architecture or business process modeling could be thought of as starting points for developing a comprehensive exchange format, also coupling these areas more tightly to EA management, which they are commonly regarded as bordering to.

Another topic, which may play an important role in future EA management may seem at first sight rather technical, but exerts a multitude of non technical implications – it is the topic of versioning EA management models, especially in respect to different dimensions of time. Commonly, temporality is implemented in the models in a *one dimensional manner*, i. e. via the notion of *as-is*, *planned*, and *to-be*, while for plans further information on the planned data are provided. Such information might be sufficient for planning the evolution of the EA, but is somewhat limited concerning traceability of changes to the plans. As an example, one might think of a plan for the EA regarding the year 2010, which might look different as-of begin 2007 respectively begin 2008. Nevertheless, it can be considered interesting, what changes between these plans have occured best complemented with information on the rationale for these changes. Such information may be useful, especially, if future plans should be adapted again, as not only the plan to be adapted, but also past planning states are directly available together with indications on the discussion that have lead to their evolution.

As with versioning also aspects of branching and merging EA management models are of importance. Resembling similarities with the software development process, the development and maintenance of a plan for the EA can be regarded to be a collaborative endeavor, especially in larger enterprises. Therefore, it might be advantageous to create and evolve these plans not as a whole, but in fragments of reduced complexity, maybe with a specific business and IT aspect in the focus. In such a distributed planning process for the EA, creating independent branches of a plan, evolving them, and merging them together into one updated plan, would be an *everyday* activity.

As of today, EA management tools do not provide sophisticated mechanisms for versioning, branching, and merging EA management models and commonly neglect the existence of different dimensions of time. There are, nevertheless, other disciplines of modeling, where aspects of collaborative model evolution are far more understood, foremost the field of content management or software engineering, where in the latter both models and code are commonly created and maintained in collaborative, distributed teams via the help of a versioning control system. We regard this to be an interesting starting point for future research projects regarding EA management tools.

Concluding this non-exhaustive list of ideas for possible topics in EA management and development, we would like to elaborate once again on the complexity of the EA. As alluded to above, the architecture shows. due to the sheer number of constituents and their interdependencies, a high *static* complexity. In handling this complexity, e. g. via visualization and collaborative maintenance techniques, the current EA management tools provide a broad range of functionalities. Nevertheless, the EA cannot be considered to be a static system only, e. g. one could think of actual information on the status of an application system (*operating vs. broken*) as more dynamic aspects of the EA. Due to the interdependencies, this *dynamic* complexity is likely to be high as well, although aspects of that kind are currently not taken into consideration by EA management tools[2]. For governing these aspects of dynamics and making them transparent to the user, we regard mechanisms for performing simulations on the EA or subsets thereof being a promising approach. With these simuations techniques complemented by methods for quantifying certain properties of the EA, likewise metrics, we see the dawn of a new EA management maturity level – further closing the gap to management areas, especially in financial management, where such methods are commonly accepted and widely used.

[2]One might have restrict this judgement a bit, as especially some of the tools originating from a business process management background provide some capabilities in this area.

APPENDIX A

List of Criteria

Id		Chapter	Question
1		Vendor data	
	1.1		Please provide the name of the tool vendor.
	1.2		Where is the headquarter of the company located? Where does the company have subsidiaries?
	1.3		When was the company founded?
	1.4		How many employees does the company employ?
	1.5		What are the main products of the company?
2		Tool data	
	2.1		Please provide the name of the tool, including version numbers.
	2.2		When was the current version released?
	2.3		When is the next major release to be expected?
	2.4		Please provide a brief history of the tool, starting with the first public release.
	2.5		Please outline the schedule for the next minor and major releases of the tool and outline the new functions in the upcoming version.
3		General tool architecture	
	3.1		Please provide an overview of the tool's architecture (e.g. three tiered architecture, thin client support, integration into portals). Please provide diagrams on the components employed in the tool.
	3.2		Please provide an overview of the tool's infrastructure requirements (hardware, operating system, RDBMS, browser - if appropriate distinguish different aspects of the tool, e.g. the thin client)
	3.3		Please provide an overview which parts of the architecture are purchased and which parts where developed by the company?
	3.4		What plattforms or database systems (e.g. DB2, My SQL) does the tool support?

4		Collaboration support
	4.1	Does the tool support multiuser work? Please provide information due to multiple reading and writing.
	4.2	What kind of synchronization mechanism is provided to support multiple user edits, e.g. locking, timestamp based synchronization?
	4.3	When locking is supported, which locking modes (e.g. shared, exclusive) does the tool support and which locking granularities (e.g. whole models, diagrams, set of entities) are distinguished?
	4.4	Does the tool provide a rights management for restricting user's access e.g. to models, diagrams or limit their editing capabilities concerning e.g. certain entities? Can users pass the rights given to them to others (with grant)?
	4.5	Does the tool support versioning of artifacts in respect to collaboration support, i.e. can a model be reverted to the status prior to changes by a certain user?
	4.6	Does the tool offer capabilities for offline working with the data, e.g. a client, from which edits can be synchronized with the repository? What kind of operations are supported on offline data?
	4.7	Does the tool support Multi-client capability to allow simultaneously access to several clients without seeing each others data?
	4.8	Does the tool support automatic notifications (especially when changing certain objects)?
	4.9	Does the tool support substitution rights (e.g. vacation replacement)?
	4.10	Does the tool support integration in corporate portals (e.g. wikis) to support collaboration? If yes, how is it implemented?
5		Internationalization / Localization
	5.1	Does the tool provide capabilities to assign a locale to a user profile? Which adaptations to e.g. the graphical user interface of the tool may be defined in a locale (e.g. date format, currency)?
	5.2	Does the tool support multi-language data, e.g. naming or description of entities dependent on the users language within one installation / instance of the tool?
	5.3	Does the tool support unicodes?
6		Integration with related domains
	6.1	Does the tool support Business Process Modeling? Which Business Process Modeling standards/notations does the tool support? EPC, BPML, BSEL, WSCI, BPEL, Other
	6.2	Does the tool support data modeling? Which data modeling standards/notations does the tool support? E/R, Crowfoot notation, IDEF1X, UML with profiles
	6.3	Does the tool support UML modeling? How many diagrams does the tool support? Which diagrams does the tool support? Class Diagram, Composite Structure Diagram, Component Diagram, Deployment Diagram, Object Diagram, Package Diagram, Activity Diagram, Use Case Diagram, State Machine Diagram, Sequence Diagram, Collaboration Diagram, Timing Diagram

7		Methodology
	7.1	Please provide information on the predefined metamodels shipped with the tool (number of classes, associations) and their compliance to frameworks for EAM, as e.g. Zachman.
	7.2	Please provide information on the number of classes (entity types) contained in the metamodels, especially of the standard or default metamodel employed for EAM.
8		Integration with other modeling tools
	8.1	Please provide information on different formats and tools, from which data can be imported into the tool, e.g. CSV, Excel, Microsoft Project. Please detail on how transformations for importing these data can be implemented or configured.
	8.2	Does the tool support for accessing data from a BPM tool by an interface? Which BPM tools are supported by which interfaces (offline, online)? ARIS Toolset (IDS Scheer), ADONIS (BOC), Corporate Modeler (Casewise),...
	8.3	Does the tool support for accessing data from a CMDB by an interface? Which CMDBs are supported by which interfaces (offline, online)? Atrium (BMC), Tivoli CMDB (IBM), CMDB (HP),...
	8.4	Does the tool support for accessing data from a Systems Management tool by an interface? Which Systems Management tools are supported by which interfaces (offline, online)? Tivoli (IBM), OpenView (HP), SMS (Microsoft),?
	8.5	Does the tool support for accessing data from a Project Management tool by an interface? Which Project Management tools are supported by which interfaces (offline, online)? Clarity (CA), Mercury PPM (HP), BW (SAP), Project (Microsoft),?
	8.6	Does the tool support for accessing data from an ERP tool by an interface? Which ERP tools are supported by which interfaces (offline, online)? SAP, Oracle,?
	8.7	Please detail on the mechanisms for synchronizing and keeping consistency with data from an external data source.
	8.8	Does the tool support a connection to workflow engines? BizTalk, Tipco
	8.9	What kind of export formats will be supported? Is it possible to export XML?
9		Administration
	9.1	How flexible is the tool due to various kinds of installation (e.g. can it handle an installation with several databases that will be distributed to different organisational units?)
	9.2	Does the tool support single sign-on and can other existing user directories (like LDAP) be used?

Table A.1.: List of Criteria

APPENDIX B

Long List

No	Name of Vendor	Name of Tool(s)	Total	Short
1	AB+ Conseil	SOLU-QIQ	24	
2	Acceptsoftware	Accep360	24	
3	Adaptive	Adaptive EAM	42	x
4	Agilense	EA WebModeler	34	
5	alfabet	AG planningIT	69	x
6	ASG	ASG Enterprise Management/Rochade	30	
7	Avolution	ABACUS	32	
8	BEA	AquaLogic Enterprise Repository	37	
9	BiZZdesign	BiZZdesign Architect, BiZZdesigner	28	
10	BOC	ADOit/ADOxx	44	x
11	BTM Corporation	BTM 360 Product Suite	26	
12	CA	Clarity	35	
13	Casewise	Corporate Modeler Suite, IT Architecture Accelerator	48	x
14	Comma Soft	infonea	35	
15	Embarcadero	EA/Studio	33	
16	Enterprise Elements	Elements Repository	25	
17	Framework Software	Structure	24	
18	Future Tech Systems	ENVISION VIP	25	
19	GoAgile	GoAgile MAP	28	
20	Hewlett Packard	Mercury Project and Portfolio Management Center	49	x
21	IBM	Rational Software Architect	46	x
22	IDS Scheer	ARIS IT Architect	68	x
23	IDS Scheer	ARIS ArchiMate Modeler	53	x
24	INOVA Engineering	MERGE-Tool	26	
25	Intelligile	Map Suite	26	
26	Knotion Consulting	SYNAP-C Solution	24	
27	LogicLibrary LogiScan	Logidex	26	
28	MEGA International	MEGA Modeling Suite	45	x

29	NetViz	NetViz	25	
30	Orbus Software	iServer for EA iServer	27	
31	Primavera	ProSight	33	
32	process4.biz	process4.biz	33	
33	Metastorm	ProVision	38	x
34	pulinco	TopEase Suite	30	
35	QualiWare	EAM Suite	34	
36	Select Business Solutions	Select Component Architect	24	
37	Sparx Systems	Enterprise Architect	32	
38	Sybase	PowerDesigner	26	
39	Telelogic	Telelogic System Architect	60	x
40	Troux Technologies	Troux	55	x
41	Visible Systems Corporation	Visible Enterprise Products	30	

Table B.1.: Vendors and Tools in the area of EA Management

APPENDIX C

Software Map Types

Contents

Software maps are graphical models of the application landscapes visualizing different aspects, which correspond to the concerns of various stakeholders and are part of an EA. The *software cartography* deals with the design and creation of software maps and makes use of concepts from (conventional) cartography.

In cartography the base map is made-up by a two- or three-dimensional space, typically a topographical map showing the earth or parts of it. A thematic map uses a topographical base map to visualize certain aspects like population density or political election results on layers. In software cartography the same layering principle is applied to visualize different aspects. Unfortunately, no two prominent topographical characteristics like longitude and latitude do exist in software cartography. Therefore, we distinguish between different software map types, which are categorized by the rules for building the base map.

To understand the elements of an software map the following three terms are of importance:

Map Symbol : Map symbols are the graphical elements of a software map. Examples are *rectangles*, *circles*, *chevrons*[a], *traffic lights*.

[a]A fish symbol, which is often used to represent a business process.

Visual Variable : A map symbol owns a set of visual variables, influencing the graphical representation of an instance. Different map symbols own different visual variables, e. g. a rectangle owns the visual variables *centerPoint*, *width*, *height*, *borderColor*, *fillColor*, *borderStyle*, *textAttribute*, etc.

Visualization Rule : A visualization rule defines visualization constraints or targets. A constraint visualization rule demands a specific relationship between map symbols, e. g. the nesting of a map symbol instance inside another map symbol instance. A target visualization rule *may* demand the area minimization of a map symbol instance. Constraint visualization rules must be fulfilled to visualizes the semantical information in a correct way. Target visualization rules should be fulfilled as good as possible to result in diagrams with *aesthetical* appearance.

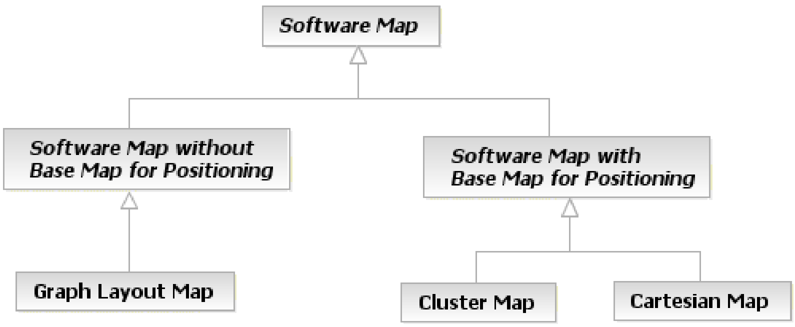


Figure C.1.: Diagram showing Inheritance of Software Map Types

The following sections introduce the different software map types (see Figure C.1).

C.1. Software Maps with Base Map for Positioning

For software maps *with* base map for positioning the coordinates of an element positioned on the base map are of importance. If the position of an element is changed relatively to the base map, its semantical meaning may be changed, too. The following two sections describe the two types of software maps in this category.

C.1.1. Cluster Maps

A cluster map is a software map, which uses logical units (named *clusters*) to build the base map and to group application systems in these units. This grouping is visualized by *nesting* the graphical representations of application systems into the symbols representing the corresponding logical unit, expressing a specific semantic relationship between the logical unit and the *nested* application systems.

Logical units are e. g. organizational units, functional areas, or (geographic) locations. According to that, the nesting may be used to e. g. visualize a *hosted-at*-relationship between a location (e. g. a computing center) and an application system.

Figure C.2 shows an example for a cluster map, visualizing which locations *hosts* which application systems. The nesting of the rectangle *Online Shop (100)* inside the rectangle *Munich* has the semantical meaning that this location hosts the application system. The corresponding visualization rule *nesting*, demanding the positioning, is explained in the legend of Figure C.2.

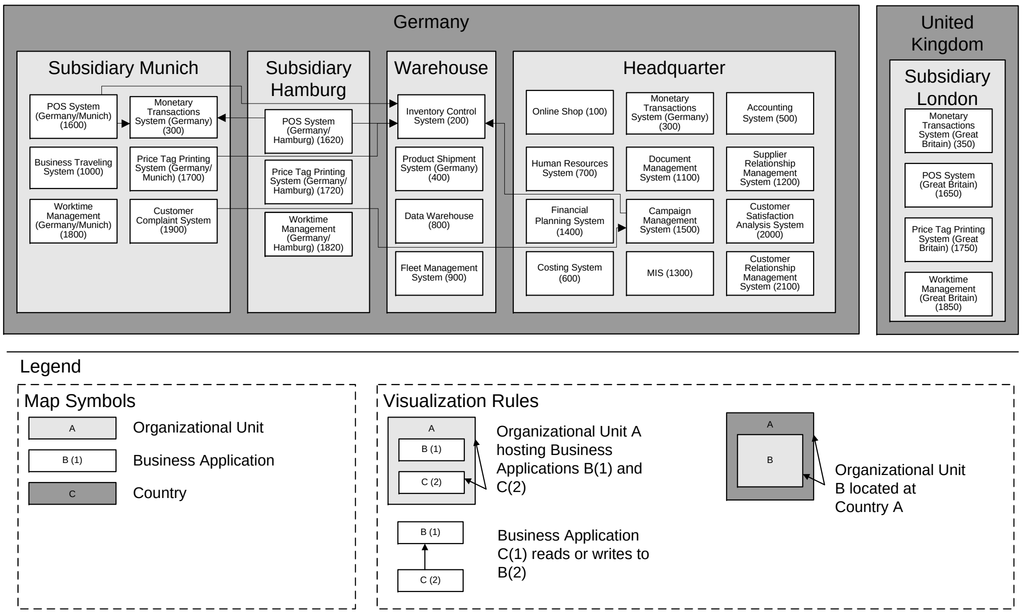


Figure C.2.: Exemplary Cluster Map

This map type does neither specify how the clusters should be placed on the base map nor how the different elements, nested in a logical unit, should be placed in relation to each other. Therefore, two different approaches for providing additional specification for positioning can be distinguished: *positioning for area minimization* and *rule based positioning*. The positioning for area minimization targets to place the nested symbols and the symbols representing the logical units in a way, which minimizes the total area required by the software map. The rule based approach uses additional semantic information to determine the positioning, e. g. by putting application systems with supplier contact left, application systems with customer contact right.

C.1.2. Cartesian Maps

A cartesian map is a software map, making use of two axes (dimensions) for creating the base map. Therefore, the base map of a cartesian map relies on two aspects of the EA for determining the x- and y-axis. The map symbol instances for the application systems are *aligned* to the elements on the x- and y-axis. Two popular types of cartesian maps are the *process support map* and the *time interval map*, detailed in the following.

The base map of a process support map has the business processes positioned on the x-axis and e. g. the organizational units or system types on the y-axis, thereby building a grid like positioning schema. An example of an process support map is shown in Figure C.3. The position of the rectangle labeled *Inventory Control System (200)* (for the application system) below the chevron *Acquisition* (for the business process) and beside the labels *Headquarter*, *Subsidiary Munich*, *Subsidiary Hamburg*, *Subsidiary London*, and *Warehouse* (for the organizational units) means that this application system supports this business process at the different organizational units.

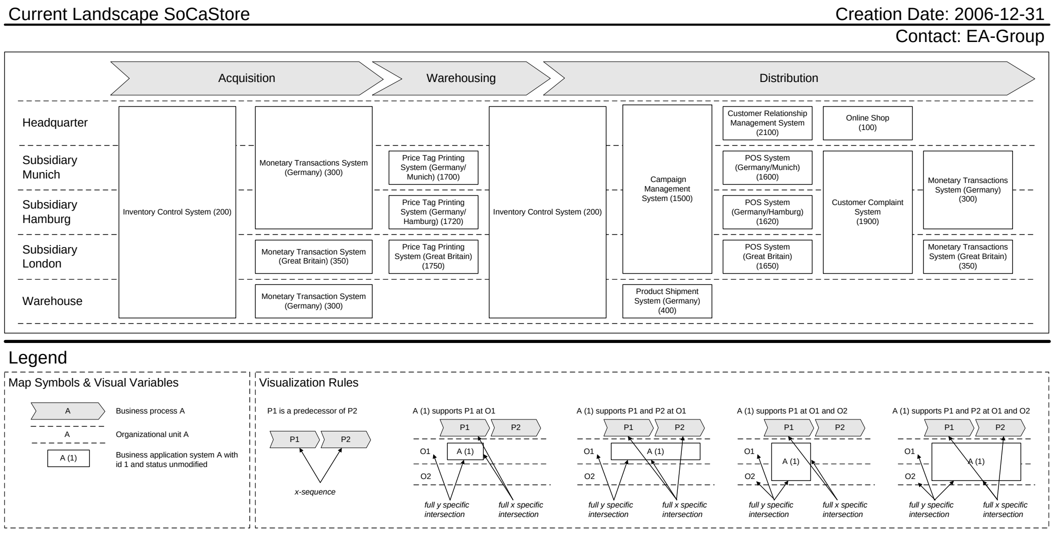


Figure C.3.: Exemplary Process Support Map

Concerning the utilization of business processes on one axis, a precondition has to be noted, demanding that the processes make up a process chain, i. e. a linearly ordered sequence. This is typically only true for high-level views on the set of processes.

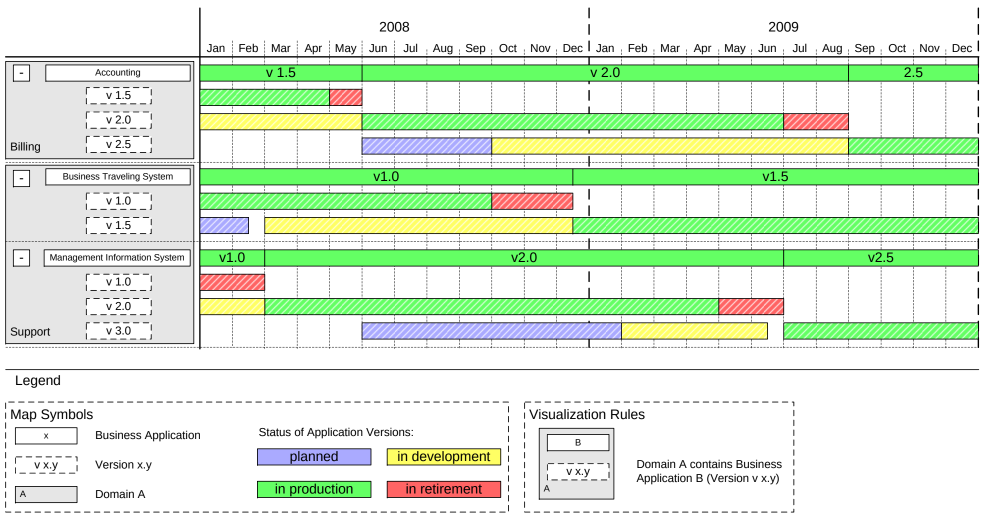


Figure C.4.: Exemplary Time Interval Map

The base map of a time interval map is similar to the basic buildup of a Gantt-diagram, with the time used for creating the x-axis. The y-axis is made up of the elements, whose development over time should be visualized. The exemplary time interval map in Figure C.4 shows the life-cycle of application systems and their versions. The rectangles positioned below the time-line and beside the application system/application system version visualizes the status of the element on the y-axis. The color coding of the rectangles is used to distinguish different status, like *planned*, *in development*, *in production* and *in retirement*.

C.2. Software Maps without Base Map for Positioning

In addition to the map types already introduced, there also exist software maps without base map for positioning. The decisions concerning the positioning of symbols on the base map are made by the map creator. As a result of this, positioning does not necessarily convey semantic information, but is used for aesthetic purposes like in a multitude of other graphical models, as e. g. in UML class diagrams [OMG05], Entity Relationship diagrams [KNS92], or representations of graphs via nodes and edges, like for Petri-Nets [JV87]. The last type of graphical models is the cause, why software maps without base map for positioning are also called graphlayout maps. Usually, this type of map is used, if the user has to create a visualization optimally suited for a certain problem, as e. g. an impact analysis.

Nevertheless, there might exist positioning rules for graphlayout maps like a rule targeting the number of lines crossing to be minimal or the symbols displayed to be equally distributed on the diagram's canvas. An example for a graphlayout map is shown in Figure C.5. For creating this map a layouting algorithm has been used, which centers a symbol representing a user-defined application system and surrounds it by symbols representing application systems, which communicate with the *central* application system. As no positioning on the base map is applicable in graphlayout maps, all information about relationships between elements has to be visualized using edges (lines).

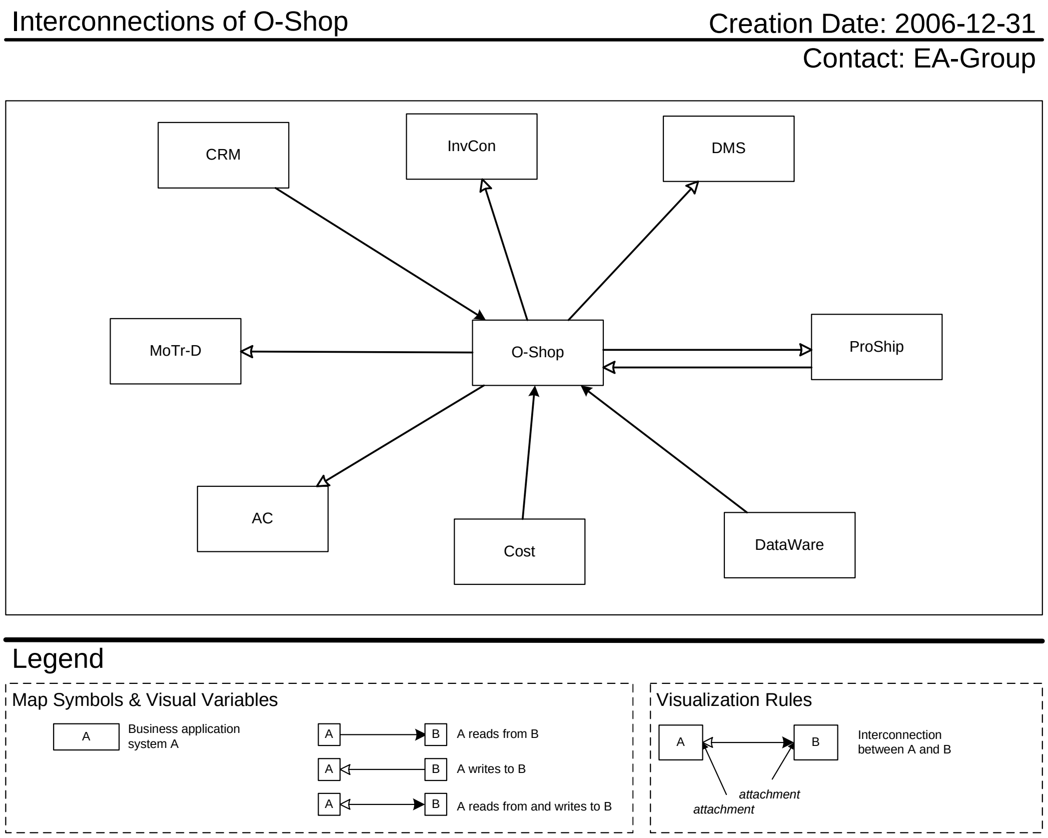


Figure C.5.: Exemplary Graphlayout Map

C.3. Layers in Software Maps

Software maps are graphical models of the application landscape and therefore visualize application systems, their attributes, and their relationships to other elements of the EA. As it is clearly not advisable to display this multitude of different aspects at the same time, software maps support, as outlined above, a principle for reducing the visualization complexity – the layering principle. According to this principle, different aspects, as e. g. operating costs of application systems or availability measures, are displayed in different layers. For any of these layers, which is not the base map, a *reference layer* exists, where the referenced symbols are seated on; this layer can be the base map. If no base map exists, due to the fact, that a map without base map is used, the lowest layer is a prominent candidate for a reference layer. Figure C.6 emphasizes on the layering principle by giving an outline, how layering can be applied.

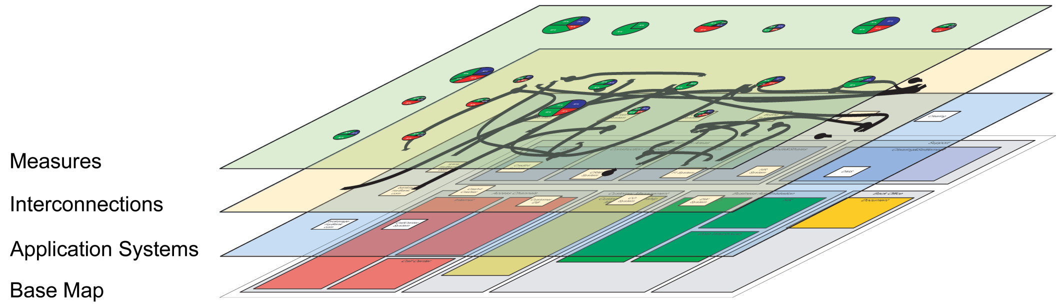


Figure C.6.: Layering principle of Software Maps

A main advantage of the layering principle is the reduction of visualization complexity, as the layers can be shown or hidden as desired by the user, thus varying the information density of the map. Additionally, the layering principle can be seen advantageous, as e. g. certain application systems are easily recognized throughout the different visualizations due to their unchanged position on the map. Therefore, the stakeholder can more easily perceive the information in a visualization, which displays additional aspects by extending a map already known to the stakeholder.

For further information on software cartography and software maps refer to [Wi07b].

Bibliography

[Bu07a] Buckl, S. et al.: *Generating Visualizations of Enterprise Architectures using Model Transformation (extended version). Enterprise Modelling and Information Systems Architectures - An International Journal.* 2(2). 2007.

[Bu07b] Buckl, S. et al.: *A pattern based Approach for constructing Enterprise Architecture Management Information Models.* In *Wirtschaftsinformatik 2007.* page 145–162. Karlsruhe, Germany. 2007. Universitätsverlag Karlsruhe.

[Bu08] Buckl, S. et al.: *Enterprise Architecture Management Pattern Catalog (version 1.0, February 2008).* Technical report. Chair for Informatics 19 (sebis), Technische Universtität München. 2008.

[DoD08] Department of Defense (DoD): *DoD Architecture Framework Version 1.5: Volume I: Definitions and Guidelines.* Department of Defense (DoD) USA, 2008. http://www.defenselink.mil/cio-nii/docs/DoDAF_Volume_I.pdf (cited 2008-03-19).

[DFH03] Disterer, G.; Fels, F.; Hausotter, A.: *Taschenbuch der Wirtschaftsinformatik.* Carl Hanser Verlag. München. 2003.

[Jo05] Jonkers, H. et al.: *A Language for Enterprise Modelling.* In (Marc., L., Ed.): *Enterprise Architecture at Work.* Springer. Berlin, Heidelberg, New York. 2005.

[JV87] Jessen, E.; Valk, R.: *Rechnernetze - Grundlagen der Modellbildung.* Springer. Berlin, Heidelberg et al. 1987. ISBN 3-540-16383-2.

[KNS92] Keller, G.; Nüttgens, M.; Scheer, A.-W.: *Semantische Prozeßmodellierung auf der Grundlage "Ereignisgesteuerter Prozeßketten (EPK)".* Technical Report Heft 89. Institut für Wirtschaftsinformatik (IWi), Universität des Saarlandes. 1992.

[Kr05] Krcmar, H.: *Informationsmanagement.* Springer. Berlin. 4th edition. 2005. ISBN 3-540-23015-7.

[La05] Lankhorst, M.: *Introduction to Enterprise Architecture.* In *Enterprise Architecture at Work.* Berlin, Heidelberg, New York. 2005. Springer.

[LMW05] Lankes, J.; Matthes, F.; Wittenburg, A.: *Softwarekartographie : Systematische Darstellungen von Anwendungslandschaften.* In *Wirtschaftsinformatik 2005.* page 1443–1462. Bamberg, Germany. 2005. Physica-Verlag.

[LS08] Lankes, J.; Schweda, C. M.: *Using Metrics to Evaluate Failure Propagation and Failure Impacts in Application Landscapes.* In *Multikonferenz Wirtschaftsinformatik.* GITO-Verlag, Berlin. 2008.

[MW04] Matthes, F.; Wittenburg, A.: *Softwarekartographie: Visualisierung von Anwendungslandschaften und ihrer Schnittstellen.* In (Dadam, P.; Reichert, M., Ed.): *GI Jahrestagung (2).* pages 71–75. Ulm, Germany. 2004.

[META02] META Group: *Enterprise Architecture Desk Reference.* META Group, Inc., 2002.

[Mic05] Microsoft Corporation: *Microsoft Operations Framework (MOF).* Microsoft Corporation, 2005. http://www.microsoft.com/technet/solutionaccelerators/cits/mo/mof/default.mspx (cited 2008-04-18).

[OGC00] Office of Government Commerce (OGC): *ITIL - Service Delivery.* IT Infrastructure Library (ITIL). The Stationery Office. Norwich, UK. 2000. ISBN 0-11-330017-4.

[OMG05] Object Management Group (OMG) *Unified Modeling Language: Superstructure, version 2.0, formal/05-07-04.* Object Management Group, 2005.

[Ro03] Ross, J. W.: *Creating a strategic IT Architecture Compentency: Learning in Stages. MIS Quarterly Executive.* 2(1). 2003.

[se05] sebis (Chair for Informatics 19): *Enterprise Architecture Management Tool Survey 2005.* Technische Universität München, Chair for Informatics 19 (sebis). 2005.

[So01] Sommerville, I.: *Software Engineering.* Pearson Studium. München. 6th edition. 2001.

[ST07] Starke, G.; Tilkov, S.: *SOA-Expertenwissen.* dpunkt Verlag. Heidelberg. 1st edition. 2007. ISBN 3-898-64437-5.

[TOG08] The Open Group: *TOGAF 8.* The Open Group, 2008. http://www.opengroup.org/togaf/ (cited 2008-03-19).

[Wi07a] Wittenburg, A. et al.: *Building an integrated IT governance platform at the BMW Group. International Journal Business Process Integration and Management.* 2(4). 2007.

[Wi07b] Wittenburg, A.: *Softwarekartographie: Modelle und Methoden zur systematischen Visualisierung von Anwendungslandschaften.* Dissertation. Technische Universität München, Fakultät für Informatik. 2007.

[Za92] Zachman, J.: *Extending and Formalising the Framework for Information Systems Architecture. IBM Systems Journal.* 31(3). 1992.

[ZIFA08] Zachman Institue for Framework Advancement: *Enterprise Architecture: A Framework.* Zachman Institue for Framework Advancement (ZIFA), 2008. http://www.zifa.com/framework.pdf (cited 2008-03-19).

List of Figures