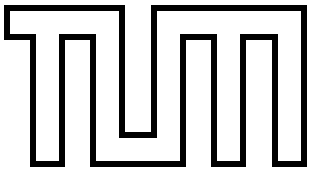


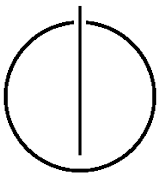
FAKULTÄT FÜR INFORMATIK

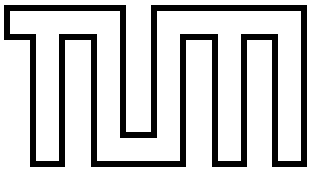
DER TECHNISCHEN UNIVERSITÄT MÜNCHEN

Masters Thesis in Informatik

Visualization of Enterprise Architecture Model Evolution Based on an Example in the Consumer Goods Industry

Nevzat Orhan





FAKULTÄT FÜR INFORMATIK

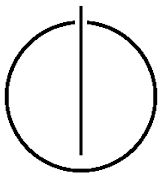
DER TECHNISCHEN UNIVERSITÄT MÜNCHEN

Masters Thesis in Informatik

Visualization of Enterprise Architecture Model Evolution Based on an Example in the Consumer Goods Industry

Visualisierung von Unternehmensarchitektur-Model-Evolutionen anhand eines Beispiels aus der Konsumgüterindustrie

Author:	Nevzat Orhan
Supervisor:	Prof. Dr. rer. nat. Florian Matthes
Advisor:	M. Sc. Sascha Roth
Date:	October 15, 2013



I assure the single handed composition of this master's thesis only supported by declared resources.

Munich, October 15, 2013 Nevzat Orhan

Acknowledgments

I would like to thank numerous people for their contribution in this thesis in countless ways. Without their ideas, assistance, encouragements and contributions this thesis would not be possible.

First of all, I would like to express my gratefulness to my supervisor Prof. Dr. Florian Matthes, for his feedbacks at strategic moments during this research.

I would like to express my deepest gratitude Bosch Siemens Home Appliances for their support during this research and especially thank to Herbert Stirmlinger for his ideas, motivation and help at any point of this research.

My thanks goes to my advisor Sascha Roth, for providing me to work in this research area and for his great suggestions and advices that help to succeed this research.

My special thanks go to my family for their undying love, encouragement and support at any condition during my research. I also would like to thank all of my friends for their positive attitudes and great friendship. Finally I would like to thank a very special person Gizem Misirli for her constant support, forbearance to motivate me for bigger achievements and movements. I am very lucky that she is a part of my life.

Abstract

Enterprise Architecture (EA) Management, started to be a significant management discipline for enterprises with the improvement of IT. Enterprises often seek ways to align enterprise structures and processes with their IT infrastructure in order to have highest productivity and efficiency in their business. EA Management is an ongoing management function to improve Business /IT alignment. In today's competitive market conditions, enterprises have high pressure to adapt their operations and architecture according to rapidly changing market conditions. Usually, EA Management enables enterprises to adapt their IT to respond for constantly changing market requirements. For decision makers, it is often important to make forecasts and predictions (what-if analysis) about their enterprises or evaluate previous decisions. Today's model repositories can be utilized to store multiple revisions of an EA model, which builds a starting point to analyze their changes and respective evolution over time. While visual means for comparing EA models would assist decision makers to evaluate future planned-states with respect to their desired target-state of their EA, today's EA tools lack mechanisms to visualize the evolution of EA models over time.

This thesis work aims to develop mechanisms to visually compare different versions of an EA model. Thereby, we develop and implement concepts of understandable and user-friendly visualizations that can communicate different versions of an EA. Our approaches are evaluated in the German consumer goods industry.

Contents

List of Figures

List of Tables

Part I.

Introduction and Foundations

1. Introduction

1.1. Motivation & Problem Statement

How do enterprises deal with their complex and highly intertwined architectures? Globalization of the world turns enterprises into more complex and coupled structures. Modern enterprises are constantly looking for solutions to handle this complexity in an efficient and less endeavour way. It is generally mentioned that in order to manage the complexity of any large organization or system, the well-planned architecture is needed. According to the IEEE Standard 1471-2000 (IEEE Computer Society, 2000) architecture is, *the fundamental organization of a system embodied in its components, their relationships to each other, and to the environment, and the principle guiding its design and evaluation* [JLtD+06]. Enterprise architecture (EA) is the variations of the definition of architecture from the enterprise point of view. According to ISO Standard 42010 the EA is, *the fundamental conception of the enterprise in its environment, embodied in its elements, their relationships to each other and to its environment, and the principles guiding its design and evaluation* [BDMS10].

During the last decade, the information technology (IT) improved tremendously. Most of the globalized companies reconfigure their structure and processes by introducing the information technology in order to have cost savings and efficiency. IT enables enterprises to discover new capabilities and make them more innovative driven. Currently, it is one of the fundamental part of any enterprise. Often, it is impossible to realize even small operations without IT. Recently, for most of the processes like accounting, sales, and distribution or any other business process lay on the IT [KBS04]. With the rising of the globalization and the constantly increasing competition, these business processes started to have a constant change. Successful enterprise should be adaptive to this constantly changing market conditions and realign its architecture dynamically.

Parallel to the improvement of IT, EA Management started to be a significant management discipline for enterprises. Enterprises often seek ways to align enterprise structures and processes with their IT infrastructure in order to have highest productivity and efficiency in their business [WBL+05]. EA Management is an ongoing management function to improve Business IT alignment. Mostly, it enables enterprises to adapt their IT to respond for constantly changing market requirements. Thus, EA Management discipline gives opportunity to decision makers to decide and give responses to changes in a time and cost effective way. Moreover, for decision makers, it is often important to make forecasts and predictions (what if analysis) about their enterprises or evaluate previous decisions. In order to have predictions or historical analysis of different EA models, multiple revisions of an EA model should be retrievable into the model repositories. Today's model repositories can be used to store multiple revisions of an EA model. Comparison of the EA models and analyzing the changes of them provides the evaluation of EA models successfully

Furthermore, visualization of different EA models would assist decision makers to eval-

uate future planned-states with respect to their desired target-state of their EA. Nowadays, EA tools can generate visualization by having the current EA models. However, there is not much work on the visualization of the evaluation of EA models or historical visualization of EA models [RM13]. Today's EA tools lack mechanisms to visualize the evaluation of EA models over time.

There is a famous phrase called "A picture is worth a thousand words" to explain the importance of the visualization in perception of data. Visualization is more efficient than words or reports when decision makers, executives or any other employees are trying to understand and gain insights from data especially when the relationship between the thousands or millions of variables is trying to be observed [CCW12]. Human brain specialized to detect the patterns, visualizations or images very easily. Thus, most of the time, efficient and understandable visualizations aid decision makers to have the right decisions in order to reach their business to success. One of the fundamental keys for the decision makers is to realize quickly and understand the huge amount of data, possible trends and important key points [Ble11]. However, usually enterprises have a huge amount of data, which makes the creation of understandable and meaningful visualizations harder for decision makers.

Finally, this thesis work aims to develop a mechanism to compare different versions of EA models and visualize them. In order to have a successful mechanism to reflect the temporal aspect of EA model, the concept will be presented including understandable and user friendly visualizations. At last, the concept will be evaluated in the German consumer goods industry.

1.2. Research Questions

In order to achieve the goal of this thesis work, the following research questions will be examined and answered.

- **What are the stakeholder requirements for visual EA model analysis and evaluation with respect to temporal aspects ?**

  Approach
  In order to examine the current practices of the stakeholder in the EA Management area, stakeholder will have an initial interview as a starting point. Moreover, the goals/concerns of the stakeholder to use an EA model evaluation will be questioned.

- **What are the existing (EA) visualizations that can be realized to communicate temporal aspects of (EA) models?**

  Approach
  Currently, there are many visualizations out both in the literature and in practice in the industry to visualize the time related and temporal data. This research will examine the present practices to find the best possible visualizations that can be used in temporal aspects of EA models.

- **How to handle previous-states of an EA in a model repository and visualize them?**

  Approach
  As it is mentioned before, today's model repositories can be used to store multiple EA models. These EA models would be compared in order to reach the desired EA model state. This will be an initial step to analyze EA model evaluation. After the successful comparison of EA models and handling the desired state of the EA model, the meaningful and understandable visualization will be realized by inspiring the examples from the literature review and practices from the current industry visualizations. Based on the requirements of the stakeholder the visualizations will be conceptualized. Stakeholder will assess all of these visualizations in the consumer goods industry.

1.3. Outline of the Thesis

The following section contains the outline this research.

Part I: Introduction and Foundations

CHAPTER 1: INTRODUCTION

This chapter presents an overview of the thesis and its purpose including the research questions.

CHAPTER 2: FOUNDATIONS

Chapter 2 covers fundamental information about the concepts and technologies that will be used during this thesis research. Moreover, it covers the literature background about the visualization and early history of it.

Part II: Design

CHAPTER 3: REQUIREMENT ANALYSIS

Chapter 3 proposes the requirement analysis part of this research. It includes the initial interview with the stakeholder and functional and non functional requirement elicitation in order to realize this thesis work.

CHAPTER 4: DESIGN

Chapter 4 proposes the design analysis part of this research. It includes the conceptual design of this thesis work in the light of the requirement elicitation.

Part III: Prototype Implementation

CHAPTER 5: IMPLEMENTATION

Chapter 5 is about the implementation details of this research in order to achieve the conceptual design.

Part IV: Evaluation and Future Work

CHAPTER 6: EVALUATION

Chapter 6 includes the evaluation of this thesis work. It has final interview with the stakeholder in order to have feedback and evaluate the work. Besides the interview, the critical feedback that is gathered during the presentations from stakeholders about this thesis

work will be reflected.

CHAPTER 7: FUTURE WORK
This chapter includes possible future extensions and research areas. Also, it gives information about critical reflections of this thesis work.

Part V: Conclusion

CHAPTER 8: CONCLUSION
Chapter 8 concludes this work by summarizing the steps to provide answers to the initial problem statement and research questions.

2. Foundations

The following chapter will discuss the foundations that include the basic knowledge to understand this thesis work. This chapter will propose an overview about the hybrid wikis and infrastructure of the EA Management tool that is used also for the implementation of visualizations. Finally, the literature review about the possible visualizations, visualization techniques and early history of visualizations will be presented.

2.1. Introduction of the System

In this sub chapter, the EA Management tool that is used for this research will be introduced. This research will use this EA Management tool comprehensively to realize the proposed scenarios.

2.1.1. Tricia Introduction

In this thesis work, the platform called Tricia is used. Tricia is a web based Java platform, which enables enterprises to run their small or big sized projects successfully. According to the company's website, creator of this tool, Tricia is a collaborative project and information management platform, which is suitable for the development teams, start-ups, agencies [web13a]. This platform is implemented by the collaboration of one of the chairs of Technical University of Munich, Software Engineering of Business Information Systems(sebis) [web13b] based on their well-rounded knowledge on EA Management and the company called infoAsset AG [web13a].

2.1.2. Web 2.0 to Enterprise 2.0

The term Web2.0 is preliminary coined by Dale Dougherty in 2004 during one of the team meeting discussion for companies O'Reilly and MediaLive International [O'R05]. Currently, famous applications like Wikipedia or Flickr are examples of Web2.0 technologies that ease the activities like collaboration, information finding, sharing, tagging [Chi08] . For instance in Germany, the popularity of Web 2.0 accrued rapidly particularly throughout the young people. More than one fourth of every Internet user above 14 years old uses Wikipedia regularly in Germany. Slightly same numbers are also observed for the other top Web2.0 applications like private social network or video portals by having private accounts [Sto10].

With the improvement of the web technology, enterprises have started to have bigger demand on the web-based technologies. Enterprises are more and more focusing on these technologies to have a collaborative platform for their internal communication and collaboration. With the improvement of Internet and Web, social software like wikis, blogs or social networks, improved significantly as creating the web content. Many enterprises

started to use web based social software systems to enable the internal communication and collaboration inside their enterprises [SS11]. Thus, companies are trying to enable the collaboration and knowledge transfer inside the companies by many content objects such as wikis, blogs or social networks [TN09] . Many Fortune 500 companies are introducing Web2.0, often called as Enterprise2.0, in order to revive active innovation, productivity and cost reduction [Chi08].

Enterprise 2.0 is a new terminology in web platform, which is used to support teams in companies by means of workflows or communication using Web 2.0 technologies. This term is coined by the Andrew P.McAfee, which aimed to label all the Web2.0 technologies that the company can build in order to enable knowledge transfer between the workers of the enterprises [Coo08]. The Enterprise 2.0 core services or properties are identified with an acronym called SLATES, which stands for Search, Links, Authoring, Tags, Extensions and Signals [McA06].

- **Search** stands for finding the relevant information inside the Intranet or web technologies of the enterprises.
- **Links** gives ability to users creates links and therefore extend the content of the enterprise data.
- **Authoring** is a service that allows users to edit or link to content according to their authoring rights.
- **Tags**, categorizes the content of the Intranet or web related technologies inside the enterprises. It enables users to categorize the content by one-word descriptions.
- **Extensions** extends the categorization and recommendation of the Intranet content for the one user according his/her behavioural actions.
- **Signals** is the awareness systems for the users. The content of the Intranet of the enterprises are generally quite big. Signal element enables users to be aware of the changes from the content of the enterprise contents.

2.1.3. Hybrid Wikis

With the rising of both Web 2.0 and Enterprise 2.0, wiki commenced to be a very demanding system, which allows technical or non-technical users to create a web content [SGW05]. This content has its own syntax called "wiki syntax", which is much easier and user friendly than HTML. Thanks to this advantage, anyone can edit, delete or add new information to the content created by wiki. However, traditional wiki have some restrictions. Although in traditional hybrid approach end users can search data and connect related information with hyperlinks, it is not possible to retrieve information in more structured way like querying information. For instance getting the list of the entire students in Germany, who are studying at technical universities with a cumulative grade of 1.0, seems not possible with the traditional approaches. Semantic wiki is another wiki type that solves this limitation. It is a powerful concept to introduce information retrieving, which is similar to querying the content of the database. However, semantic wiki has some restrictions while editing the content of the wiki pages because of the unfamiliar syntax and modeling

concepts. Furthermore, semantic wiki currently does not support for good collaboration inside the enterprises because the non-expert users may have difficulty to use semantic wiki systems [MNS13]

Hybrid wikis uses the benefits of both simple wikis and semantic wiki and produces a powerful concept like having both structured and unstructured content [MNS13]. Figure 2.1 shows an example of a hybrid page with specifying structured (Figure 2.1 (2)) and unstructured (Figure 2.1 (1)) content. Every user can easily edit, add or delete data from the unstructured content without knowing any syntax information. This unstructured content may include writings including picture, video and email attachments. It is a similar approach just like in traditional hybrids. Other than unstructured content, Hybrid wikis has structured content with including key value pairs [MNS11]. This key-value pairs are named as a term called attributes in the Hybrid wikis. Thanks to this property, end user can easily define some attributes about the wiki pages. In the given example (Figure 2.1), there are attributes called "Title" or "Project"(Figure 2.1 (3)) and values of these attributes. (Figure 2.1 (4)).

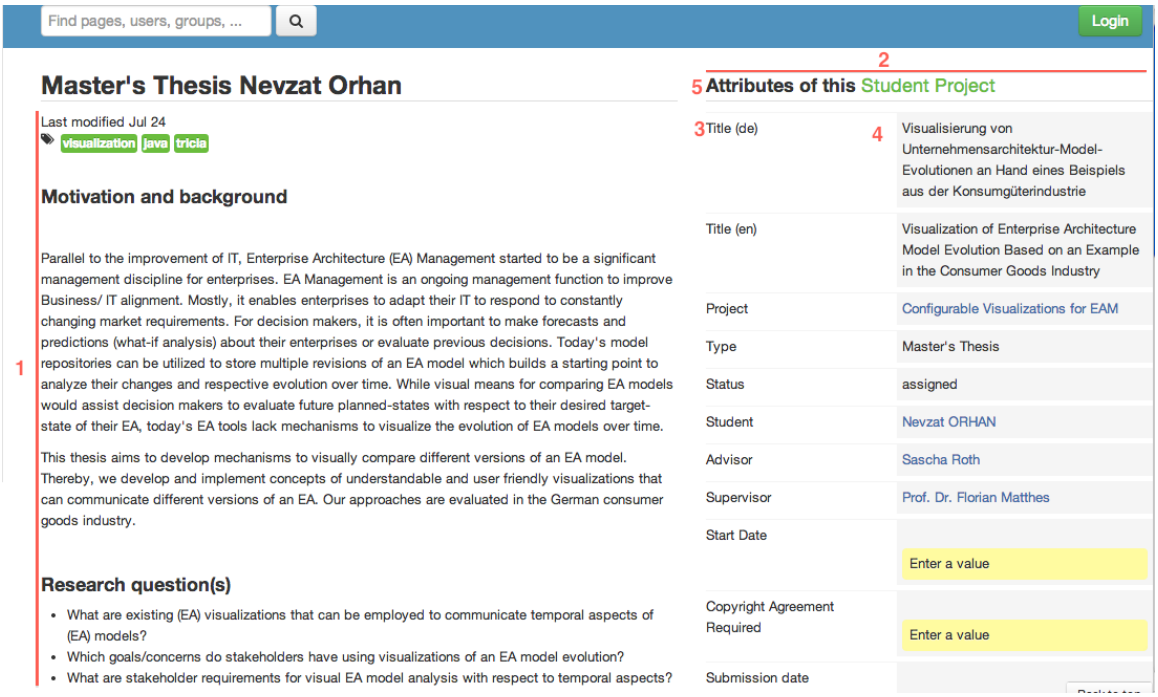


Figure 2.1.: Sample wiki page.

Type tags is another important concept in Hybrid wikis. Type tags enables users to specify the object that is being described in the wiki page [MNS11]. In other words, type tags is very a concept in hybrid wikis to determine the sort of the wiki page. In the given example "Student Project" (Figure 2.1 (5)) is a sample name for the type tags. By this way, some additional attributes can be created automatically using the system as a recommendation. The system detects the type tags and recommends mostly used attributes for the given type tags. This gives inexperienced users to stay in the scope while creating the new page by forcing users to create recommended relationships. Moreover, experienced users can

still expand the wiki pages by introducing new attribute names and type tags. Type tags is also useful to categorize pages under the given type tags objects. The user can easily access these categorized pages by just clicking on the assigned type tags.

2.1.4. Tricia as Hybrid Wiki

Tricia is a Java based commercial Enterprise 2.0 software with the realization of concept Hybrid Wikis. Tricia is an extended wiki mechanism by using the Enterprise 2.0 technology with mechanisms called classification, linking, consistency checking, visualization of wiki pages [MN11]. In overall, Tricia is a powerful tool with strong features like supporting the collaboration for the enterprises by enabling networking and personal and team blogging and linking properties.

Figure 2.2 provides an IT infrastructure of Tricia platform. The following information is provided by the Ph.D. dissertation by Christian Neubert [Neu12] and [MM12].

Tricia platform supports both cloud and desktop applications, which gives a system substantial flexibility. Tricia offers *Hypertext Transfer Protocol (HTTP)* (S) to enable web users to access to the Tricia server, by including *Asynchronous JavaScript* and *XML* (AJAX)-style asynchronous interactions and accessing the file directories are ensured by protocol like *File Transfer Protocol (FTP)*. *Representational state transfer (REST)* enables third party applications to access and interact with Tricia Server. Tricia has an active access control system which enables the system to define different roles for the system to determine different read and write access for the content objects (like wiki pages) [MN11]. The system administrator is a special user who can configure Tricia server by using the Tricia Configurator. Tricia platform needs *Java Run Time Environment (JRE)* on Windows and Linux, the Lucene full text search engine, and database server. Tricia has internal data source as MySQL [1] and Oracle [ORA13]. Moreover, for the development and testing purposes, it supports HSQLDB [2] database [Neu12].

Figure 2.3 represents an overview of Tricia data modelling framework concepts that are introduced in details in [Neu12] and [BMN10]. In the following paragraphs, some of the most relevant concepts for this thesis work will be discussed.

Asset is an abstract superclass of **Entity** and **Mixin**, that enables these two concepts to have a capability of having features. Feature is another important abstract concept assigned for properties and roles.

Entity is the representation of the Tricia's domain objects like Wiki or WikiPage. Entity has an internationalized label and a name. Every data object is inherited from the entity objects. Tricia consists of inertial repository in which every data objects should be stored. To achieve this, each entity should define a persistence schema instance which will generate a schema of the given data object inside the Tricia persistence repository. Figure 2.4 [3] shows a class diagram, which explains the relationship between the entity and the persistence entity. In Tricia, every PersistentEntity has its own unique id string. There is another concept similar to id called unique identification number (UID), which is the combination of id and the type of the PersistentEntity object. Since it is not possible insert an element

[1] `http://www.mysql.com/`; visited on September 18th 2013.

[2] `http://hsqldb.org/`; visited on September 18th 2013.

[3] `https://bitbucket.org/sascha.roth/tricia-visualizations-nevzat`; visited on September 23th 2013.

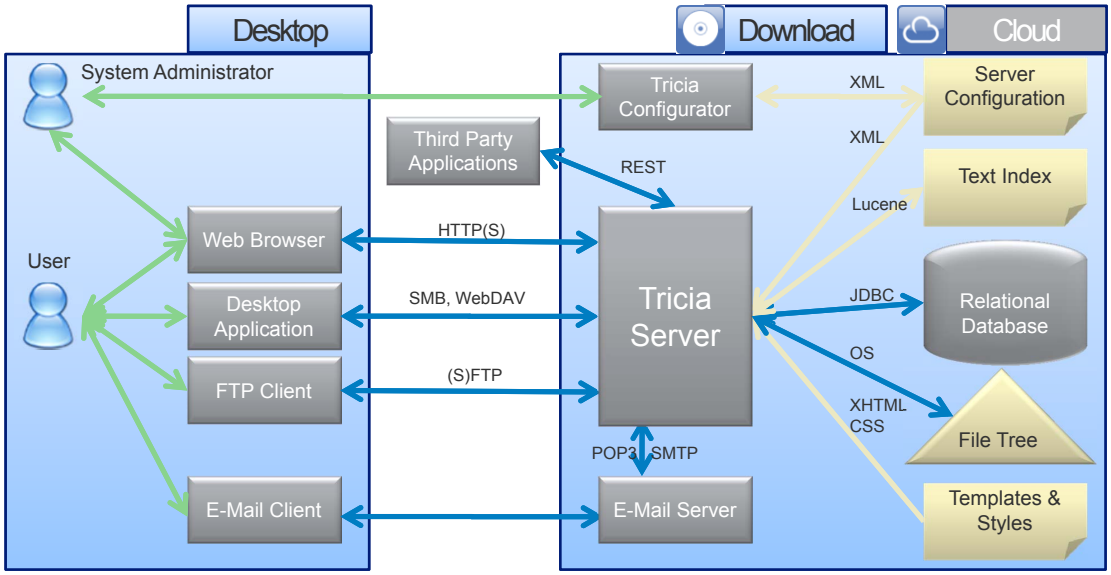


Figure 2.2.: General overview of Tricia.

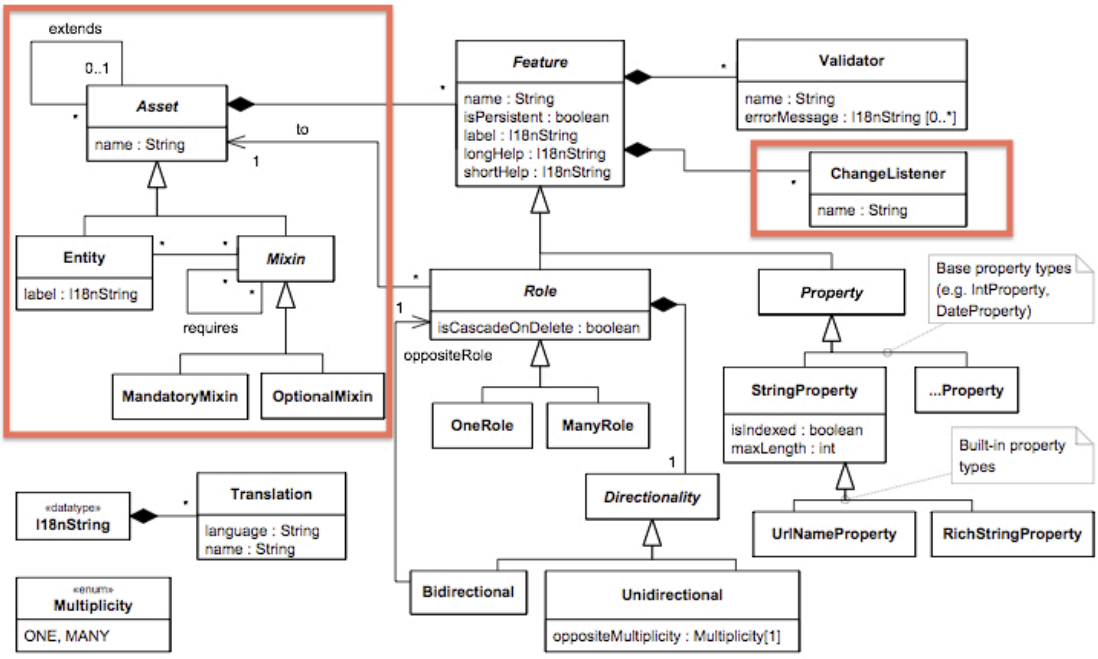


Figure 2.3.: Data modeling framework of Tricia [Neu12][BMN10].

with the same unique id into the relational database tables, the insert mechanism should be extended in such a way that it can ensure continuous data import to the persistence database. This extension will be explained in details in the upcoming chapters.

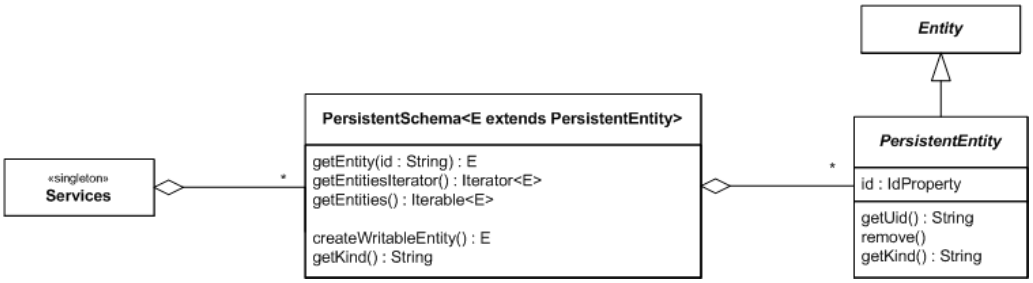


Figure 2.4.: Entity-Persistence layer Uml class diagram

In complex systems, it is often important to define particular concept or functionality and reuse it by different data types. Java enables this by using inheritance, but only can deal with one superclass since Java does not allow multiple inheritance [4]. Mixin allows programmers to have better modular solutions in Java by solving multiple inheritance restriction. Mixin consists of two different mixin types called **MandatoryMixin** and **OptionalMixin**. MandotaryMixin is assigned to the entity and cannot be removed in run time, whereas OptionalMixin can be removed in run time [BMN10]. There are several Mandotary Mixin used during this thesis work named, Versionable, Linkable and Hybrid.

- **Versionable** mixin allows Tricia to keep the history of the assigned entity. This mixin assigns statically when there is a need to watch for specific entity.
- **Linkable** mixin defines the outgoing and incoming links to the specified entity.
- **Hybrid** mixin helps to retrieve the Hybrid object of the specified entity which contains the structured part of the entity

ChangeListener is used to notify the system about the data model changes. It has a specified name and it is registered on a feature [BMN10]. In overall, ChangeListener appraises the system when the properties of the feature of the asset have changed.

Figure 2.5 illustrates a logical class diagram of simplified data model of Tricia. This class diagram aims to show the conceptual behaviour of the data object in the Tricia. According to the data model, there are terms like Space, Page, Tag Type, Links etc. In the following sections, the details of these attributes will be explained.

Space is a concept, which behaves like a container. Mostly it is beneficial for grouping the content under particular concepts. Moreover, space objects give capability to the system to set identical elements into one structure and make more precise and neat data representation. In other words, for instance, a space can be named as a "Students" which consists of many student wiki pages.

PersonSpace is a type of space which contains the set of users of the Tricia system. These users are important part of the system used for the active access control to the data content. The memberships or access rules are based on the PersonSpace of the Tricia.

[4] `https://bitbucket.org/sascha.roth/tricia-visualizations-nevzat`; visited on September 18th 2013.

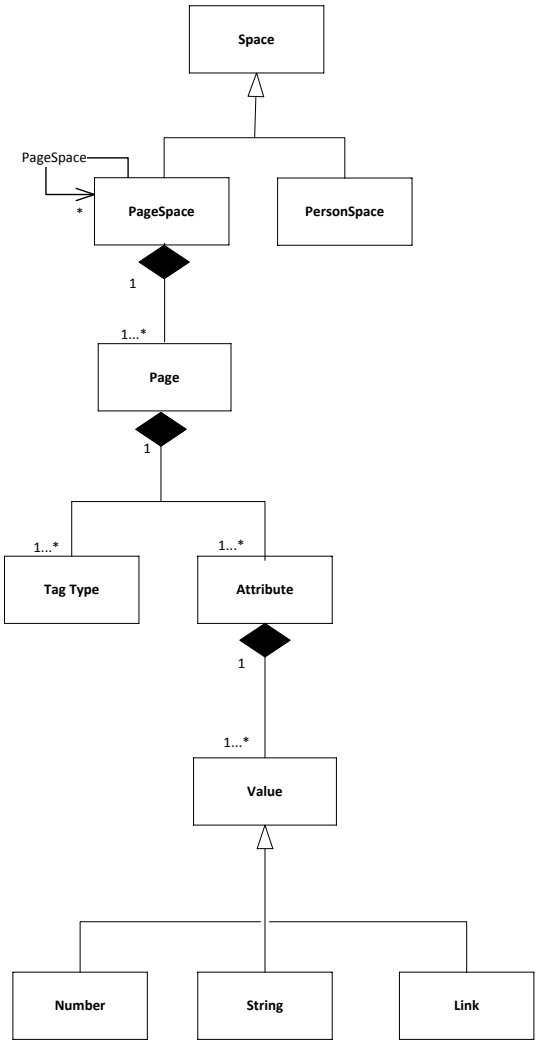


Figure 2.5.: Tricia hierarchy data model.

PageSpace is another type of space which contains the set of pages or content object with the same type of PageSpace. In other words PageSpace is the container of the Page objects.

Page is a content object, which can be also called as a wiki page that is the one of the strength of the Tricia. It consists of both structured and unstructured content. Structured part of the page object consists of structured attributes, which is a combination of attribute - value tuples. The values of these tuples can vary from Number objects to any Link objects. Unstructured content contains tag and unstructured data. Unstructured data contains any content including text or any embedded media object like visualizations or videos.

Link is a special type in Tricia to show references and relations to different pages or object inside the system. There are two different link types in Tricia called internal links

and external links. Internal links point the pages or object inside the application, on the other hand external link point the pages or objects outside the application. Incoming links can be either internal or external links pointing from the current page or object to outside pages. Moreover, outgoing links can be either internal or external links pointing from the current object to outside objects [Neu12].

Tag is another important concept of the Hybrid Wiki pages. There are two types of tags in Tricia. The first one is **Tag Type**, which belongs to the structured content part of the wiki page and explained comprehensively in this chapter before. Second one is normal tag, which is the part of the unstructured content of the hybrid wiki page. Tags have many functions and motivations in the system. They are used for organizational purposes, contribution and sharing, expressing opinion etc. [Zol07]. Many well-known websites like del.icio.us or YouTube or Flickr intensely using tag based system [HK07]. Tricia uses tag clouds thanks to comprehensive ability of Enterprise 2.0. In Tricia, tags are used generally to explain the characteristics' and properties of the unstructured content of the hybrid wiki pages.

2.2. Introduction to Tricia Visualizations

Today's enterprises have many different applications in place. These applications have long lifecycle time and the total number of them is increasing rapidly. EA Management enables enterprises to have an efficient management of huge amount of applications with high degree of coupling. Thus, appropriate EA management tool support has great importance for all the stakeholders in the industry in order to have a better documentation and visualization to support users of the enterprise thus better management on their EA [BDM+08]. According to the literature [ELSW06], there are two distinct problems with the visualization of enterprise data or structure inside EA management tools. These problems are, lacking of automatic generation of visualizations in EA management tools and definition of semantics of visualization in ambiguous terms, respectively. In the same literature, it is stated that these two problems are investigated because of manually creation of visualizations.

According to the literature [HMRS], manually creation of visualization becomes a cumbersome effort when there are many other different dynamically changing information models in one enterprise. In order to solve this problem, there should be a generic solution approach, which enables automatic visualization generation, based on constantly changing information models [HMRS]. Figure 2.6 shows a conceptual framework that is proposed to solve these problems. The concepts that are proposed in the figure are becoming more and more abstract as it goes to bottom direction.

The explanation about the elements of the conceptual framework is based on the literature [HMRS] and [SMR12]. **Data model**, which is an actual data, can be retrieved by querying from the data source. After that, information model is formed which is a schema based on the actual data source. According to Lee, *Information model is a representation of concepts and the relationships, constraints, rules, and operations to specify data semantics for a chosen domain of discourse* [Lee99]. **View data model** constraints the results of data model query and creates new information by aggregating the existing values. **Abstract view** model forms a definition of visualization and matches it with the information model,

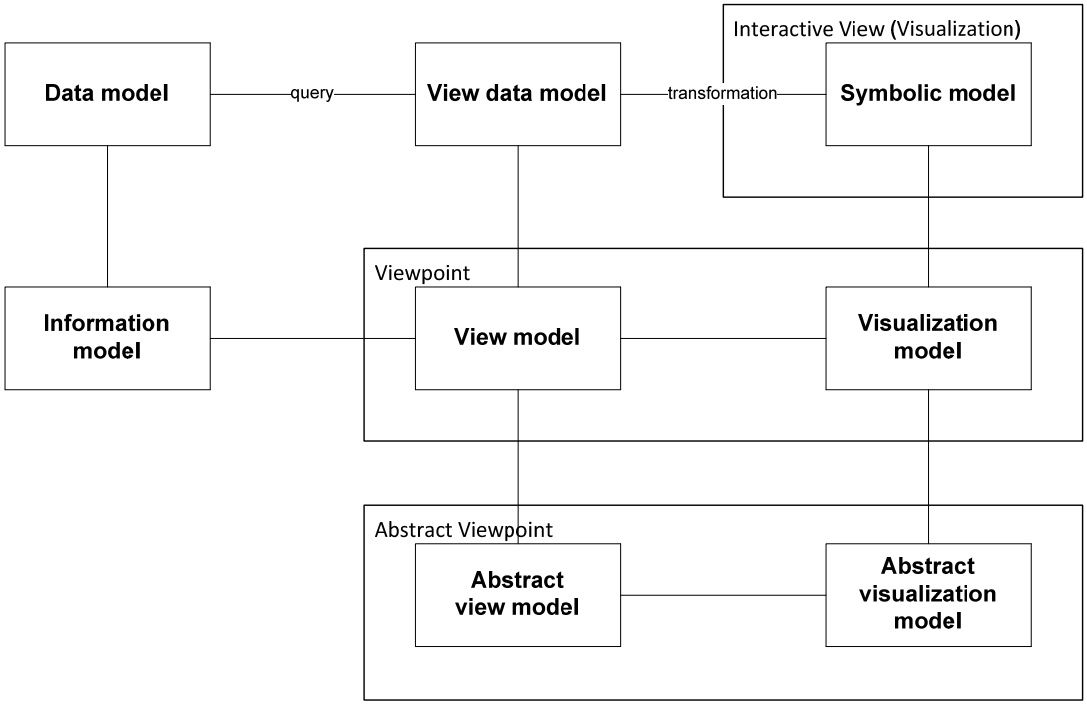


Figure 2.6.: Conceptual framework for automatic visualization generation [HMRS].

which is provided by the View model. **Symbolic model** is the instance of the primitive figures like circle, rectangle, etc.. **Visualization model** is the definition of the visual figures that was briefly introduced in the symbolic model. In other words, visualization model is the description of the symbolic model like a circle with a red dot inside. **Abstract visualization model** is used to describe more complex composition of elements. In order to create the visualization based on the structural definition of the Abstract view model, the Abstract viewpoint binds Abstract view model to the Abstract visualization model. After the successful binding of the visualization, the end user can configure the Abstract Viewpoint in order to decide on the visualization elements design issues. This allows analyst to specify the primary shapes, design and configuration of the predefined visualizations.

Tricia is using Raphael framework in order to generate visualizations and interactions in the web browser [HMRS]. Raphael framework is a JavaScript library that enables programmers to create easily customized vector graphics and interaction on the web browser [5].

The implementation details of this framework will be discussed later in the implementation part. More detailed information about the conceptual framework of Tricia's automatic visualization generation can be found in the literature papers [HMRS] [SMR12].

[5] `http://raphaeljs.com/`; visited on September 18th 2013.

2.3. Visualizations

Starting with the improvement of computer and related technologies, the development of information visualization and importance of it triggered rapidly. However, the presence and usage of the information visualization extends to ancient times [Spe07a]. Firstly the following will give some sample visualization that was discovered long before the invention of computer and its related technologies.

According to the Tufte, Figure 2.7 represents the one of the earliest statistical graphics that was created by Micheal Florent van Langren in 1644. The visualization is representing the various different estimated distances between two cities, Toledo and Rome. It represents twelve different estimation of astronomers and cartographers and the estimation of Langren by using his own method. Langren estimated the position of Rome, spread out the values between 22 and 25. Although the estimations were quite failure, the visualization itself was sophisticated enough according to its concurrent visualizations [Tuf97]. Figure 2.9 is another well-known time-honored visualization that was developed by John Snow who was trying to find the reason of the Cholera broke out in London. Thanks to his map, John Snow found out the particular region of the London where did the cholera break out. Thanks to the systematic work of Snow's Map, the specific reason for the cholera was found [Tuf86a]. Figure 2.8 is another classical famous visualization that has formed by Charles Joseph Minard (1781-1870), the French engineer. The visualization shows the number of the soldier that is varying according the time during the Russian campaign of 1812 [Tuf86b]. This visualization represents the population of the army with the geographical coordinates to represent the location of the army. Moreover, this map shows places the army is passing through including the date and direction information.

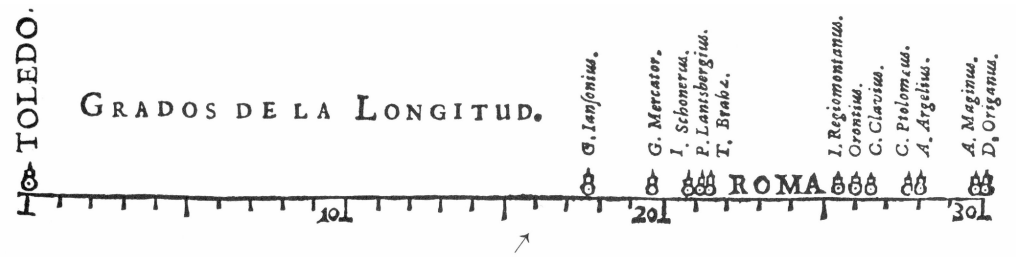


Figure 2.7.: Visualization of Micheal Florent van Langren

William Playfair (1759-1823) is an important personage of the early history of visualizations. Playfair invented line graph, bar chart, pie chart and circle graph that are today one of the most widely used visualization in all areas of life. Figure 2.10 shows one of the Playfairs inventions, which brings together pie, circle and line chart [YVMF06].

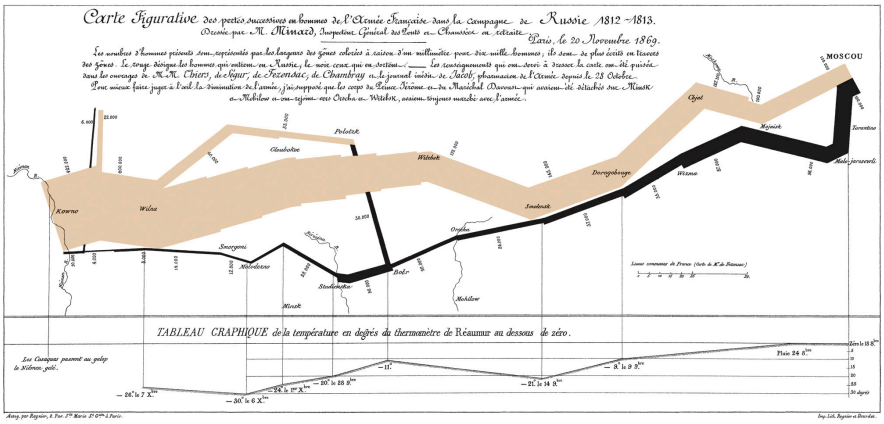


Figure 2.8.: Minard's flow map

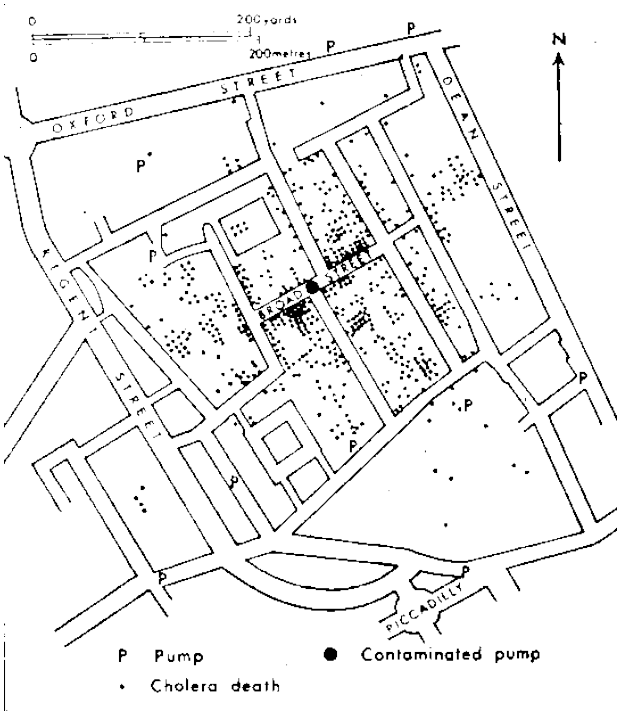


Figure 2.9.: A map of cholera deaths in London, 1840s

In the literature, there are plenty of visualizations from the very basic ones to the very sophisticated ones according to their needs and usage area. There are many sample classifications of the visualization based on some taxonomy, which will be discussed later. However, the following visualizations are not grouped considering any taxonomy, but they are compared based on their visual similarity and usage area. After introducing the following visualizations, they will be grouped under the Keim visualization techniques and data type classification [Kei02].

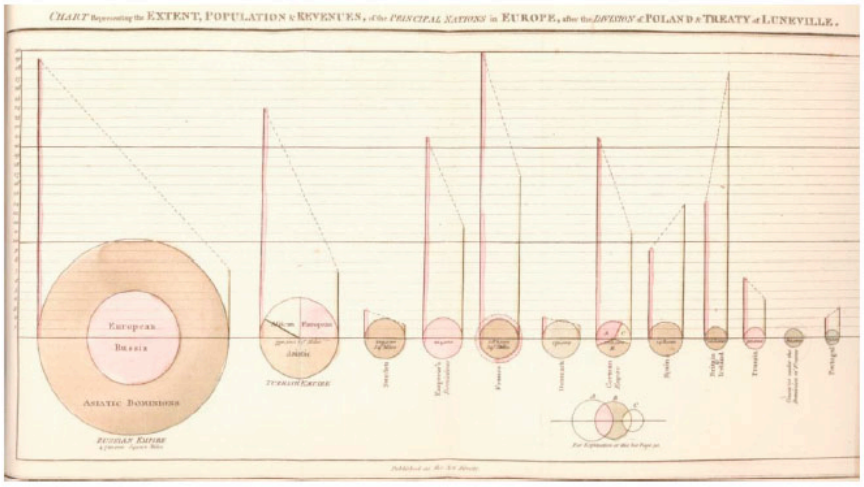


Figure 2.10.: The first graphical representation of Pie-Line-Bar chart

- **Bar chart** [YVMF06] is one of the mostly used and well-known visualization that was discovered and firstly used by William Playfair. Bar chart is commonly used for visualization of large data volumes because of its familiarity among people. It is generally used to compare different attributes. **Pixel bar chart** [KHDH02] is derived from the bar chart in order to show data directly inside the bar chart by using pixels. In order to achieve this, each pixel inside the bar chart column represents one data value. Another extension of bar chart is called **stacked bar chart** [AR12], which is heavily used in order to discover the relationship between different categories or attributes. In stacked bar chart, along the x-axis or y-axis, the small box structures stacked on top one another.

- **Histogram** is another well-known visualization, that is very similar visually to the bar chart. However, they have some minor visual differences. For instance, bar chart is used to compare different attributes, but histogram is used to represent the distribution of one attribute [Rob12]. **Sparkline histograms** [TW10] are extensions of histograms, which offer a compact representation of the data set. They are used for visualizing large histograms set by using smaller graphics called sparkline.

- **Pie chart** [YVMF06] was firstly used and discovered by William Playfair. **Polar-area diagram** [Lew06] is one of the first extensions of the Pie Chart, which was firstly used by the Florence Nightingale. It is used for showing mortality rate at different periods of time. Owing to the limitations of the pie chart like difficulty in estimating relative size of similar slices and toughness of comparing different partitions, **spie charts** [**?**] was discovered. Spie charts provide the efficient comparison of different pie charts. Similarly, **fourfold display** [Fri94] is discovered in order to develop more efficient pie chart comparison. Fourfold display, offers an efficient comparison of dichotomous attributes for at least two populations. **Organic pie chart** [MÖ8] is another visualization build open on the pie chart. It looks like a pie chart, but the final representation depends on the data. **Circle segment** [AKK] is introduced in

order to visualize the large amounts of data. It has a similar layout as Pie Chart however, the functionality of them is totally different. In order to visualize different dimension of the data, the circle is divided into the number of segments that includes the given data. Moreover, by having the pixel-per-value technique, large amount of data can be displayed by assigning each data to one pixel.

- Traditionally, time series are used to analyze trends (observing rapid changes in a long-term period), seasonal variation (observing the unemployment rates according to the seasons) and some other cyclic changes (constantly repeating events or trends) [Cha03].

  Time series data can be visualized either by using line graphs or stacked bar chart. However, both of these concepts are limited in terms of visualizing the changes and trends especially when the data gets bigger. The concept called stacked graph is discovered, in order to solve the limitations of line graphs and stacked bar chart. **ThemeRiver** [HHWN02] is used to detect the trend between the set of themes or topics. It gives an overview of thematic changes over a set of topic or documents. It uses a river metaphor to represent the themes or topic by mapping different colors to each "currents". **Stacked Graphs** [BW08] can be seen as an improvement of ThemeRiver that relies on traditional stacked graph visualization. It was discovered by Lee Byron in order to detect the trends in music listening. **Horizon Graphs** [HKA09] called as "space-efficient time series" that improve time series visualization in the use of space and precision. The horizon graph enables users to view negative and positive values in one view with a mechanism called filled line charts. In Horizon graph, positive and negative values are separated by different colors, which are filled up from the x-axis. Even, this filled line chart mechanism is layered to increase the efficiency.

  Other than stacked graphs there are some other novel concepts in order to visualize the time series data analysis. **SparkClouds** [LRKC10] enables the detection of the trends between multiple tag clouds by using the Sparkline [BDM+08]. **DiffMatrix** [SLKS12] is another approach to show trends in a matrix-based interactive visualization. As the number of line graphs increases, the overlapping problem occurs. In order to solve this problem, DiffMatrix offers a novel approach by offering a matrix like visualization. Moreover, by employing different approaches like dual lines, diff lines, diff area and diff heat map, DiffMatrix enables the user carefully and accurately compare two different time series. Likewise DiffMatrix, **Data Vases** [TR09] offers another novel approach to visualize multiple time series. It is based on the kite diagram, which is formed by the help of the configuration of the line graph. The aim of the visualization is to visualize and compare high amount of time series in a less amount of space.

- In order to visualize two quantitative variables, one of the well-known technique namely, **Scatter plot** [Sel10] can be used. Scatter plot consists of x-y axis and a point for each x-y match. Different shapes and even color-coding can be used in order to increase the level of visualization. **Star coordinate** [Kan00] is another type of plot visualization, which can be seen as the improvement of Scatter plot. Differently from the scatter plot, a star coordinate aims to compare and visualize multidimensional

data set. **Parallel Coordinates** [ID90] is another solution for visualizing multidimensional data set. It is based on a system called parallel lines and the mapping between these parallel lines. **Table lens** [RC] is another way of visualizing a large amount of tabular multidimensional data.

- **Tabular data** is often used in almost every field from education to business. Classically, tabular data is visualized in a table representation, which is not very efficient because of space limitations and not offering any visual manner. **CircoSonic (Circos)** [Ngu12a] is offering a novel approach to visualize a large amount of tabular data. It is a circular graph that enables the comparison of pairwise relationship in a 2D table. Contingency table is a special kind of table to report the categorical data. **Contingency Wheel** [AGMS11] presents an interactive visualization to visualize contingency table.

- Tree visualization is one of the widely used visualization to show the hierarchical or tree structures. There are two kinds of visualization techniques of tree visualization called, space filling technique and non-space filling technique or node- link diagram. Space filling technique has a space-efficient design by having the special rectangular and radial layout. On the other hand, the node-link diagram is suffering from the space inefficiency, but still one of the most widely used visualization to show hierarchical and tree based data [WGK10].

  Radial Tree [SBB+03], **Hyperbolic Browser** [Spe07b], **Cone Tree** [RMC91] are examples of node-link diagram trees. All of these trees have different layouts, but the idea of them is basically the same. They consist of different nodes which has assigned links to make a connection between different nodes.

  Treemaps [JS91] [CMS99] is one of the very well known tree visualization with the space filling technique. Node-link diagram is very nice visualization in a manner of aesthetic and usability when the issue is to visualize the small tree structure. However for large trees, node-link diagram is limited because it is unable draw this structure in a limited display spaces. Tree-Maps finds an efficient solution to scalability problem of the node-link diagram by omitting explicitly drawn internal nodes. The Tree-Maps is formed by nested rectangles in order to show the hierarchy. According to this, root of the tree big rectangle container and the children of it represents as inner rectangles. The rectangles are encoded with different sizes and colors according to the quantitative variable that they are representing.

- **TimeLine** is a special type of graphical visualization to express historical events clearly. It is a useful way of visualizing temporal data. **TimeLine Trees** [BBD] aims to visualize the hierarchical data with the evaluation of them by using timeline visualization. In order to accomplish this, TimeLine Trees is combining both the tree and timeline visualization. **Time Slice** [ZDFB12] is another novel approach to visualize the evaluation of temporal data by using a web browser. It allows comparison of timelines with different attributes by using filter mechanisms. **PlanningLines** [AMM+08] is another novel approach in timeline visualization, which aims to address the temporal uncertainties during the project management and planning.

- **Maps** have been one of the widely used visualization throughout the history of humankind. Up to 17th century, map was one of the main visualization. One of the first world map was drawn in the early ancient time, which presents in the books of Herodotus [Fri05]. Afterwards, many other map visualizations had drawn in order to discover the different parts of the world. These maps were generally used to get knowledge about the geographical places and were only paper based. With the improvement of the information technology, maps are transferred to computers, and they start to use benefits of computer technologies such as Web2.0. Google Maps[6] or Bing Map[7] are one of the well known visualizations that are heavily used in the society. These maps can easily show every detail of the geographical information. For instance, Google Maps launched a new feature called Street View by May 2007 to resemble the geographical information in street-level without loosing any map content [Vin07]. Currently, digital maps do not only show geographical information, but they also illustrate some other information like pictures, advertisements, navigation information.

In order to understand the visualizations deeply, the taxonomy framework is often needed to understand the purpose, application scope and usage areas of the given visualization [QZP03].

Shneiderman defined a visualization taxonomy framework based on the data type. This taxonomy is based on data types 1-, 2-, 3 dimensional data, temporal and multidimensional data, tree data and network based data [Shn96]. Moreover, Keim classified visualization in three different criteria called visualization technique, interaction and distortion technique. Keim subdivided data type classification into following seven different categories. These are called, one-, two-, multi- dimensional data, text and hypertext, hierarchies and graphs, algorithms and software. The following table represents the Keim's data classification type based on the visualization introduced before.

Table 2.1.: Visualizations according to Keim's Data Type Classification.

Data Type	**Visualization Name**
1-Dimensional Data	Theme River, Time Series
2-Dimensional Data	Bar Chart, Scatter Plot, Histogram, Pie Chart
Multi Dimensional Data	Parallel Coordinates, TimeLine Trees, Planning Lines, Table Lens, Star Coordinate
Text and Hypertext	Theme River, Rank Explorer [SCL+12]
Hierarchies & Graphs	All kind of tree visualizations

Second, Keim classifies the visualization techniques as standard 2D/3D displays, geometrically transformed displays, icon-based displays, dense pixel displays and stacked displays. The following table represents the visualization techniques with the sample visualization name.

[6] `https://maps.google.de/`; visited on September 18th 2013.
[7] `http://www.bing.com/maps/`; visited on September 18th 2013.

Table 2.2.: Visualizations according to Keim's Visualization Technique.

Visualization Technique	**Visualization Name**
2D Display	Bar Chart, Scatter Plot
3D Display	Cone Tree, Star Coordinate
Geometrically Transformed Displays	Data Vases, Parallel Coordinate, Time Line Trees
Icon-Based Displays	Chernoff-Faces [MER00]
Dense Pixel Display	Pixel Bar Chart, Tree Maps, Circle Segments
Stacked Display	Stacked Bar Chart, Horizon Graphs

Moreover, Keim also classified visualizations according to the interaction and distortion technique that are divided into Interactive Projection, Interactive Filtering, Interactive Zooming, Interactive Linking and Brushing [Kei02].

Part II.

Requirement Analysis and Design

3. Requirement Analysis

3.1. Requirement Elicitation

There are many techniques that can be applied during the requirement elicitation. In order to gather the requirements, the initial interview and constant direct observations held during the process. Afterwards, functional and non-functional requirements were gathered by the help of the stakeholder.

3.1.1. Initial Interview

Interviewing is one of the fastest and most simple way of gathering information from the stakeholder [Som07a]. A structured interview, which consists of predefined open questions [Som07a], is one of the interview type to collect information about the stakeholder, their expectations and needs.

As a first step, structured interview conducted with one of the responsible employee who is responsible for EA management. The purpose of the initial interview was to get an overview about the current practices of EA management of the stakeholder, the arising problems and limitations and possibilities to improve their EA management system in the context of EA model evaluation.

The questions start from the most essential questions of their practices starting with their background information. Interview questioned the limitation of the stakeholders' system and possible solutions that are presently in practice.

1. **How many years does the sponsored company has EA management?**
   Answer
   The EA Management system has taken place since 2006 with a centralized team.

2. **For which purposes are you using EA Management?**
   Answer
   The EA Management is used to align IT/Business functions and technology standards of the enterprise.

3. **Which departments are currently using EA Management system? What are their roles?**
   Answer
   The architecture department of the company mostly uses EA Management system. The results are mainly presented to the IT-Board management to get a general overview about the enterprise.

4. **Do you have any tool support for your company's EA management?**
   Answer

Yes, since 2007 EA management has a tool support. This tool has developed by the internal teams.

a) **Have you used other tools for the EA management before?**
Answer
Yes, for the evaluation purposes time to time some other tools in the industry have been used. However, none of them has been chosen by the enterprise for the practical usage.

i. **If you have used other tools, what did the reason for not using these tools? What were the limitations?**
Answer
The biggest limitation was lack of flexibility of these tools. Generally speaking, the model of these tools did not fit to our enterprise model. Furthermore, the architecture should have built according to the model of these tools. Because of that, these tools were not preferred.

ii. **How long have you used these tools?**
Answer
These tools were only evaluated. Because of the limitations, they were not selected by the enterprise.

5. **What are the current approaches in sponsored company for EA Management?**

a) **How do you enable contribution to the EA Management?**

i. **How do you manage the content management?**
Answer
IT-Product managers mainly fill up the content. It is very decentralized and based on collaborative approach.

ii. **How hard does to change the content in your current EA Management?**
Answer
Both experienced and non-experienced users can easily change the content in EA management tool. It is very easy to use and user friendly thanks to wizard - based form. It does not need any other further guidance or training. It is very intuitive and structured.

b) **How do you transfer data to your EA Management system?**
Answer
We do not need any data transfer mechanism. When we need to transfer big amount of data, the administrators of the system update the servers with basic scripts.

i. **Is there any automatic import mechanism?**
Answer
In our current approach, we do not need an import mechanism. However for our research, we will need an import mechanism to transfer our data to the Tricia.

c) **How do you create the visualizations?**

i. **Do your visualizations independent from your information model?**

A. **Are they reusable and can be applied to another information models?**
Answer
The visualizations are usable in case of reconfiguring the visualization. In order to have visualizations, there should be a correct data binding. After the correct data binding, the visualization is rendered and desired report is realized.

ii. **Do you have any means of visualization configuration tool?**
Answer
Yes, we are currently using SQL Reporting Services.

iii. **Can non-experienced people create visualization without having any help?**
Answer
Every experienced or non-experienced end user can customize the reports and the visualization without having any assistance.

iv. **How easy does to create new visualization in your system?**
Answer
Very Easy - **Easy** - Moderate - Hard - Very Hard

d) **Is there any change awareness system?**
Answer
There is no change tracking system available in the tool. However the system can aware whether it is updated or not, but it cannot realize what has been changed in the system.

6. **What are the limitations of your EA Management system?**
Answer
Currently our EA Management system does not have any change tracing system, thus no historical comparison of different EA models.

7. **Why does the sponsored company need evaluation of their EA models?**
Answer
In order to have a historical visualization, we need to have a version mechanism and evaluate different historical EA models. Thus, this is the preparation for the historical visualizations.

8. **Why does the sponsored company need historical visualizations?**
Answer
The biggest need is to see the trends and changes in the company. This will help us to verify our decision and see the structure of the company in a better and confident environment.

9. **What was the solution to the limitations of the sponsored company before?**
Answer

The enterprise did not have any to the solution for this limitation. We are expecting this master thesis content will bring solution to the limitations.

3.1.2. Functional Requirements

After conducting the initial interview with the stakeholder and having constant observation through the enterprise, the functional requirements are gathered. Functional requirements [Som07b] are statements to understand the react of the system according to the particular input, situation and behavioural moment. In overall, the functional requirements should tell what should system does and does not. In order to document the requirements, Volere requirements specification template [RR06] [vol13] is used. This template groups the requirements under the requirement type, based on their similarities and purposes. This helps to detail the requirements during the documentation.

While forming the functional requirement template, especially for the visualization requirements the "possible questions" section is added to clarify the need for the particular visualization. While creating the visualization, it is often important to ask questions to express to which problem dies the visualization addresses? [HBO10]

Table 3.1.: Requirement Type 1.

Requirement Type	1
Topic	Import Mechanism

Requirement	1	**Requirement Type**	1
Description	The system should have an automatic import mechanism in order to import the enterprise data in a fast way to Tricia		
Fit Criterion	Administrator of the system should import the data without any error.		
Priority	**High** - Medium - Low		

Table 3.2.: Requirement Type 2.

Requirement Type	2
Topic	Version Mechanism

Requirement	2	**Requirement Type**	2
Description	The system should take several versions of the EA models		
Fit Criterion	The administrator of the system should create multiple accurate version for the desired date		
Priority	**High** - Medium - Low		

Requirement	3	**Requirement Type**	2
Description	There should be an easy interface for end users to take a version		
Fit Criterion	The administrator of the system should create multiple accurate version for the desired date		
Priority	**High** - Medium - Low		

Table 3.3.: Requirement Type 3.

Requirement Type	3
Topic	Visualization - Historical Visualization

Requirement	4	**Requirement Type**	3
Description	The system should visualize big and highly coupled data in a understandable form		
Fit Criterion	The end users should understand the big data in a short time		
Priority	**High** - Medium - Low		

Requirement	5	**Requirement Type**	3
Description	The system should visualize the several different states of EA instances		
Fit Criterion	The administrator of the system should easily create accurate historical and forecast visualizations		
Priority	**High** - Medium - Low		

Requirement	6	**Requirement Type**	3
Description	The historical visualization should be easy to understand and does not need any further assistance		
Fit Criterion	The end users should understand the visualizations in a reasonable time without having any external assistance		
Priority	**High** - Medium - Low		

Requirement	7	**Requirement Type**	3
Description	The system should show the relational attributes obviously and clearly		
Fit Criterion	Decision makers should see the dependency of each product, location etc. clearly		
Priority	**High** - Medium - Low		

Table 3.4.: Requirement Type 4.

Requirement Type	4
Topic	Technology Standard (Visualization Concept)

Requirement	8	**Requirement Type**	4
Description	There should be a visualization to state the whole technology infrastructure and standard technologies of the organization and compare them with different versions easily		
Possible Questions	Which technologies are standard for different type of technology categories? How many different technologies are standard per category? How to compare differences between different versions of technology categories?		
Fit Criterion	Decision makers of the organization should clearly understand the whole technology infrastructure and standard technologies of the organization and easily identify the differences.		
Priority	High - **Medium** - Low		

Table 3.5.: Requirement Type 5.

Requirement Type	5
Topic	Distribution of Technology Standards

Requirement	9	**Requirement Type**	5
Description	Distribution of the standard technologies and usage area of them should be visualized		
Possible Questions	How are the standard technologies used inside the organizations? • How is the evaluation of these technologies? How many IT-Products do not use standard components? What is the distribution of the most and least used standard technologies?		
Fit Criterion	Decision makers should be clearly identify the different technology standards used in the organizations and the distribution of them The trend evaluation should be clear		
Priority	High - **Medium** - Low		

Table 3.6.: Requirement Type 6.

Requirement Type	6
Topic	Security and Compliance (Visualization Concept)

Requirement	10	**Requirement Type**	6
Description	There should be a visualization to show and compare the security and compliance attributes of the IT-Products of the organization in different states		
Possible Questions	What kind of data is used in IT-Products? Which IT-Products work with confidential data or information? Which IT-Products are accessible from the Internet? • By which product groups?		
Fit Criterion	Decision makers should quickly observe the specific IT-Products according to their security and compliance levels. Decision makers should easily observe the differences between different version and should make judges about the security of the organization		
Priority	High - **Medium** - Low		

Table 3.7.: Requirement Type 7.

Requirement Type	7
Topic	Technology Lifecycle (Visualization Concept)

Requirement	11	**Requirement Type**	7
Description	There should be a visualization to show the lifecycle of the technologies and reflect the changes of lifecycle in different states		
Possible Questions	Which Technologies will run out of the maintenance in the next year? When do the replacement activities have to be planned to start/finish?		
Fit Criterion	Decision makers should easily aware of the expiring technologies in the organization		
Priority	High - **Medium** - Low		

Table 3.8.: Requirement Type 8.

Requirement Type	8
Topic	Dependency Analysis (Visualization Concept)

Requirement	12	**Requirement Type**	8
Description	There should be a visualization in order to show the dependencies between the different items of IT Products and systems.		
Possible Questions	Which Systems / IT-Products provide data to particular selected systems? Which Systems / IT-Products get data from the specific system? Which kind of technologies will be used? How critical are these interfaces?		
Fit Criterion	Decision makers should investigate data provider System / IT-Products easily and should make propositions about them		
Priority	High - Medium - **Low**		

Table 3.9.: Requirement Type 9.

Requirement Type Topic	9 Processes and Locations (Visualization Concept)

Requirement	13	**Requirement Type**	9
Description	There should be a visualization to observe relations between processes of the organizations and locations of them and able to compare them according to the different states of versions		
Possible Questions	Which IT-Product supports which process in a specific location? How many IT-Products support the same process? • What is the lifecycle of these products? • Which ones are planned and are planned to be shutdown? What changes can be expected in a location within the next years? Which IT-Products are used in which country? How will be the portfolio of a country change in the next year?		
Fit Criterion	Decision makers should clearly observe the locations and processes		
Priority	**High** - Medium - Low		

Table 3.10.: Requirement Type 10.

Requirement Type Topic	10 Trend of changes (Visualization Concept)

Requirement	14	**Requirement Type**	10
Description	The evaluation of the whole portfolio should be observed in the system through visualizations		
Possible Questions	How has the portfolio changed during the last periods? • Status Changes (Amount of productive / planned / shutdown) • Technology usage • Portfolio complexity etc. • Observable relevant changes?		
Fit Criterion	Decision makers should be able to quickly discern the differences between the updated portfolios		
Priority	**High** - Medium - Low		

Table 3.11.: Requirement Type 11.

Requirement Type	11
Topic	Comparison plan / actual data

Requirement	15	**Requirement Type**	11
Description	The system should contain a visualization to express the current portfolio situation		
Possible Questions	What is the current portfolio?		
Fit Criterion	Decision makers should examine the current portfolio solution expeditiously.		
Priority	**High** - Medium - Low		

Requirement	16	**Requirement Type**	11
Description	There should be a visualization to show the different aspect of the specific portfolios		
Possible Questions	What is the ratio of unplanned / planned / shutdown changes?		
Fit Criterion	-		
Priority	**High** - Medium - Low		

Requirement	17	**Requirement Type**	11
Description	The overall portfolio should be visualized per version in order to give an overview of the portfolio for the specific dates		
Possible Questions	What are the status changes? • Amount of productive / planned / shutdown • Technology usage? • Portfolio complexity etc.		
Fit Criterion	Decision makers should make a meaningful projections and judgments about the organization		
Priority	**High** - Medium - Low		

Requirement	18	**Requirement Type**	11
Description	The changes in the portfolio should be examined easily by the visualization		
Possible Questions	What have been the reasons for these unplanned / planned / shutdown changes?		
Fit Criterion	The reason of updates or changes should be obvious to the decision makers in a small response time		
Priority	High - **Medium** - Low		

Requirement	19	**Requirement Type**	11
Description	There should be a visualization to express the evaluation of portfolio changes		
Possible Questions	What is the evaluation of the portfolio changes during the last periods??		
Fit Criterion	Decision makers should easily perceive the portfolio changes for a specific time interval		
Priority	High - **Medium** - Low		

3.1.3. Non Functional Requirements

- **Look and Feel Requirements**
  - Appearance Requirements
    * The visualizations should be well understandable and have a business sense and standards.

- **Usability and Humanity Requirements**
  - Ease of Use Requirements
    * The system should be understandable, straightforward to the decision makers.
    * The data import mechanism should be clearly described.
    * Creation of visualization should be easy.
    * Data filtering or search mechanism should be usable and produce usable results.
    * The visualization creation mechanism should be straightforward in a way that users will not create wrong visualizations.
  - Personalization and Internationalization Requirements
    * The pages or visualization pages should be configurable enough for the personalization in order to readability and comfort for the end users.
  - Learning Requirements
    * The system should be easy to discover and should be productive in a short time.
    * There should not be a big effort to learn the visualizations.

- **Performance Requirements**
  - Speed and Latency Requirements
    - The visualizations should be formed in a reasonable time (Considering the time out of browsers).
  - Precision or Accuracy Requirements
    - Automated import mechanism must definitely give correct and accurate result.
    - Versions must be formed accurately.
    - Visualizations must give correct and accurate results.
  - Capacity Requirements
    - The big data of the stakeholder must be handled by the import mechanism and visualizations.

4. Design

After stating the problem context with introducing the basic concepts and the requirements, this chapter will propose the solution design of the thesis work. This design chapter aims to produce ideas and solutions for the proposed problems of the stakeholder.

4.1. Solution Design

The following sub chapter will discuss the solution design in the light of requirements and initial interview with the stakeholder. This thesis work uses software prototyping, in order to demonstrate the concept and experiment of the possible design options and end up with developing ideas about the problem and its solution. In the requirements engineering process, a software prototyping may help with the validation of system requirements [Som07c].

Figure 4.1 illustrates the proposed solution design in the light of the requirements and the initial interview. According to the solution design, the outer data source, in this example SQL application database, will store the enterprise data. Then, this outer data source will be automatically imported to the EA repository. The import mechanism should ensure that the import is continuous so that the stakeholder, in this case administrator, can import the changes in the application database in different time intervals. Based on the data schema inside the EA repository, EA models should be created in the EA Management tool. The user can update these EA models and changes are updated to the EA repository. Thanks to these changes, the stakeholder should create different versions of EA models and understandable historical visualizations in order to compare different versions. For the creation of the historical visualizations of EA models, delta analysis should be accomplished which aims to compare differences and reach to the desired EA models versions.

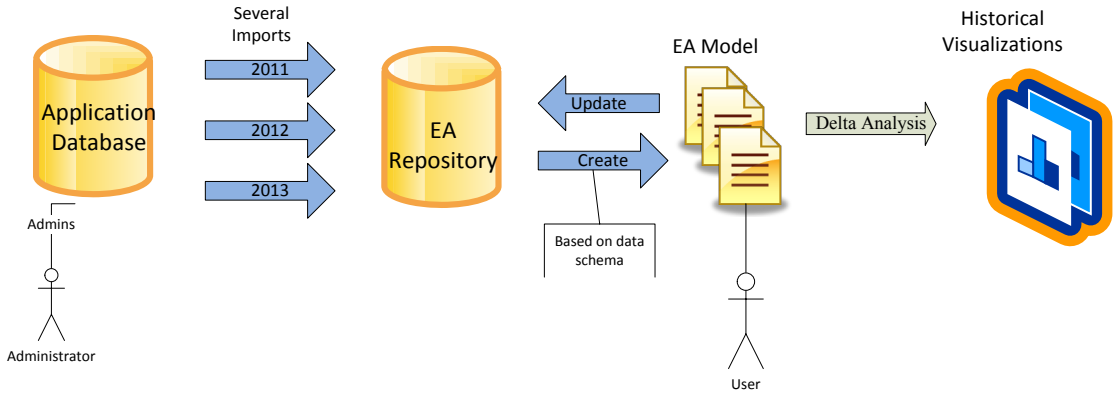


Figure 4.1.: Prototypical solution design

To realize and find the suitable visualizations for the context of this research, requirements of the stakeholder are used. Thanks to this, there is a possibility to work on real industrial problems and real solutions. As the requirements are divided into parts called "requirement type", the visualization in solution design will be shown according to this grouping. Requirement types from 4 to 11 are looking for new visualization solutions to the problems tat are stated in the requirement definition. Among these requirement types, requirement types from 4 to 8 will be introduced under the solution design, based on the literature. All other requirement types are implemented and will be introduced under the solution design part based on the implementation.

The following sub chapters will propose the solution design for the visualizations. Solution design for the visualization consists of two different parts. The first part is mainly based on the literature and looks for the best practices both on the literature and in the industry. First part contains the visualizations based on the examples in the literature. Some of these visualizations were updated based on the requirements of the stakeholder. Second part, contains the visualization that is mostly implemented in this research. In this chapter, only the design and appearance of the visualizations will be explained. The implementation details will be given in more detail in the upcoming chapter.

4.1.1. Conceptual Visualizations for Solution Design

Requirement type-4 (Table 3.4) aims to express the items of the whole IT infrastructure and to give detailed information about the technology standards. In order to visualize this requirement type, currently the stakeholder is using tabular data, which includes the categorical and discrete information. Tabular data is used often to represent the categorical data. At this point, it seems that stakeholder made a right decision by using table representation. However, the data representation itself is not categorical, but it hierarchical. Thus, table representation is not proper and convenient for visualizing hierarchical data. When the level of data increases, it becomes harder to visualize it with the table representation. As the data size increases, it becomes more difficult to observe the column row relation.

Hierarchical data commonly can be represented with a form of a tree. Each item in

the dataset represents a node, which can be a collection of other items as child nodes. In other words, this relationship consists of data, which has child relations. This relationship structure forms a tree network [Wil09]. For instance, according to the sample data, one technology standard may support different IT-Technologies and these IT-Technologies may have different versions. As a tree structure, one updated tree visualization is used which is inspired from the *SmartArt* of the Microsoft Office software [mic13]. The reason of using this visualization is the high popularity and familiarity of visualization and the software product among the business employees. In order to give a business sense, this visualization is decided to be used as a conceptual idea. Figure 4.2 is an example of this visualization according to the sample data. This visualization can represent the whole technology standards portfolio easily. Color-coding[MMN88] has used to distinguish between the different types of information by means of using different colors. The legend helps users to understand which part they are observing. Moreover, by the timeline, the user can choose different states or versions and can compare any changes between different states.

Figure 4.2.: Tree Structure visualization.

As it is stated before, **requirement type - 4** (Table 3.4) is visualized with using tabular data by the stakeholder. However, the proposed solution came out to visualize this hierarchical visualization with a tree structure visualization. However, both tabular visualization and tree structure visualization are problematic when the size of the data grows excessively both in columns and rows. FanLens [XST08] discovered to address the scalability problem in the literature. The FanLens is suitable for visualizing categorical and hierarchical data in a more compact and space efficient way. Furthermore, it supports more than one dimension by including angle information, different level of slices and color-coding

etc. (Figure 4.3)

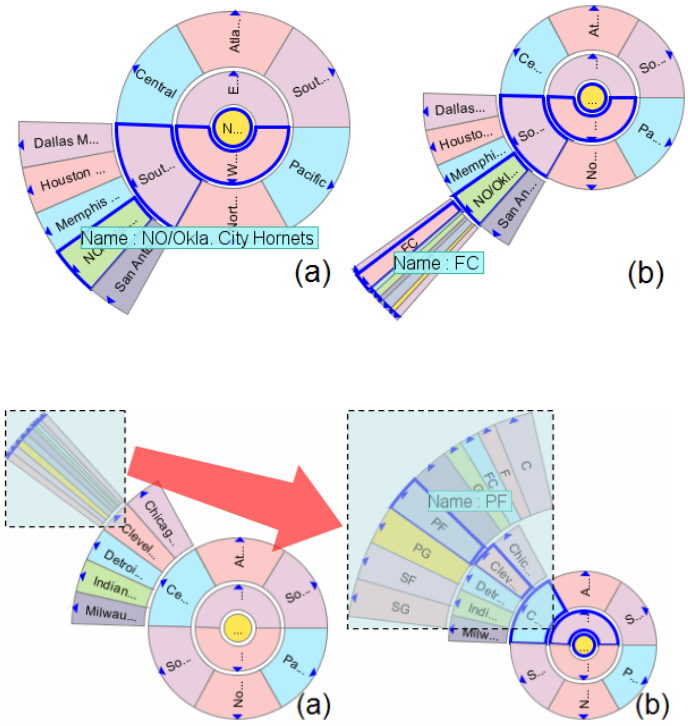


Figure 4.3.: FanLens [XST08].

FanLens offers an intuitive solution to the problem statement by having the incremental layout, zooming and picking. According to the literature, incremental layout in FanLens has two principles. These are high-level start-up and expanding/collapsing mechanism. As it is observed in Figure 4.3, FanLens does not visualize the entire data. It is user dependent to visualize the particular part of the data by expanding/collapsing mechanism. Besides, if data gets tearing for one specific layer, then normally the dimensions will get smaller, thus it decreases the readability of data. FanLens solves this limitation by procuring the zoom functionality, which magnifies the sweep angle of the focus. In overall, the FanLens would be very suitable visualization because it enables compact and intuitive representation for requirement type - 4.

Requirement type-5 (Table 3.5) intends to express the distribution of standard technologies. In other words, to make a common description, this requirement can be generalized as expressing the distribution of one element in EA management. To be able to visualize this kind of data, the stakeholder has been using a pie chart. Currently, this visualization satisfies the stakeholder in a suitable context. However, this research more focuses on the evaluation of temporal data, pie chart has limitation of envision the comparison of different versions.

Pie chart is a suitable visualization to show a definite partition of one element like standard technologies. However, when it comes to comparison of two different states of one standard technologies or comparison of different standard technologies, pie chart has

some limitations. A Spie Chart [Fei03] proposes a visualization, which visualizes different pie charts in a layered form. Spie Chart enables the comparison of different pie charts. Figure 4.4 offers one example of the Spie Chart by using the sample data.

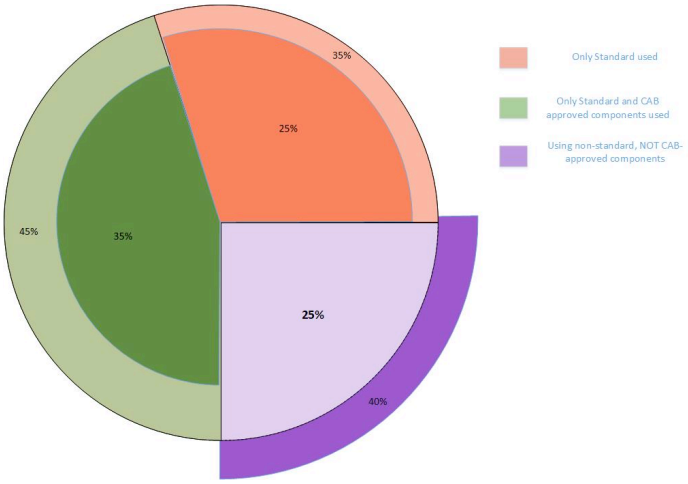


Figure 4.4.: Spie Chart [Fei03] representation .

As it is stated before, Spie chart enables comparison of two different pie charts by converting them into a representation of one big pie chart. In order to achieve this, the first pie chart is placed in the base or bottom, then the second pie chart partitions overlapped into the first one. By changing the radius of the superimposed partition, Spie Chart enables the efficient differentiation between the layers. Figure 4.4 is used to show the two different states of distribution of standard technologies according to the sample data. Other than the radius of the partitions, color-coding is used to help the differentiation between both partitions and pie charts. Different colors show the different attributes, which is clearly stated in the legend. Moreover, the opacity of the colors shows the differentiation between the pie charts. Lighter colors show the base partitions and the dense colors show the superimposed partitions. By this visualization, it is easier and efficient to compare two different versions. The literature [Fei03] discusses the limitations and criticism of this visualization. The first criticism is, if the focus is the derived partition or superimposed pie chart, then this visualization is not suitable to perceive it. However, this visualization is suitable only for comparison of the partitions, which is the real aim of the visualization. Another limitation of the Spie chart is, it can solely visualize the data, which are identical or in the same scope. Although this visualization has a bunch of limitation, it is prosperous in comparing different pie charts.

The visualization called CircleView is examined deeply to address another solution for **requirement type - 5** (Table 3.5). CircleView offers a solution to visualize the temporal data. As it is stated before, requirement type - 5 is looking for a solution to visualize efficiently different states for distribution of one item in EA Management. CircleView offers an improved solution by combining hierarchical techniques such as tree maps and circular layouts like Pie Charts and Circle Segments [KSS04]. Since in the context of the requirement type - 5, the hierarchical representation is not needed, the circle view is updated according to the needs of the available requirement. Figure 4.5 represents an updated

version of the CircleView available in the literature. The visualization seems like a circle, which consists of several layers and a hole in the middle. The group or the scope of the elements is signified in the hole. As it is stated before, this visualization consists of several layers. Each layer from inner to outer shows one particular version. Thanks to this visualization, end users can easily compare different versions or states of their temporal data.

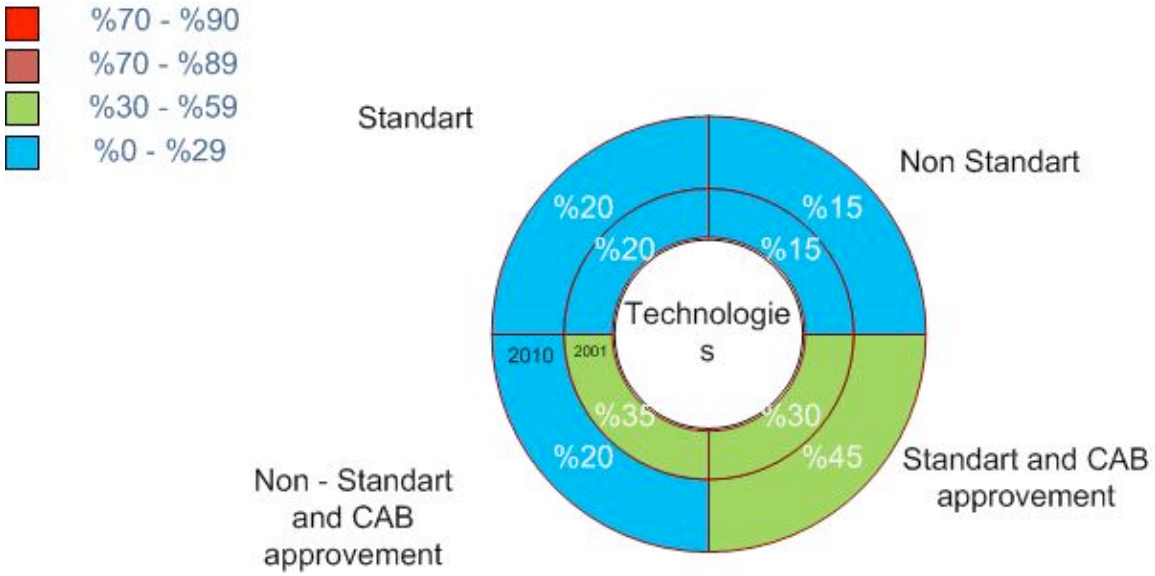


Figure 4.5.: Updated Circle View.

Radar chart is an analytical tool that is standard visualization and graphics tool in commercial products like MS Excel [1], MS Word [2] or MS PowerPoint [3]. Because of its high familiarity, it is often chosen as an analysis tool in the industry. According to the literature [MM99], radar chart has two substantial appendages which enables visualization of different indications that are easily recognizable even by the non-expert users. Thanks to its sophisticated surface area, it enables multiple dimension comparison. The name of the visualization comes from its affinity to the radar screen [MM99]. The figure 4.6 is a sample radar chart that enables to compare different versions. Thanks to color-coding in radar chart, it is very intuitive to compare different version. In the radar chart, each color can be reckoned as a different pie chart. Thus, it is a very useful visualization that can be used instead of Spie chart and CircleView because of its simplicity and representation style.

In the literature, there exists a comparison among Spie charts, target plots and radar plot by using an example data in health care industry [SLF+11]. According to the results, the Spie chart comes out to be the best visualization among two others in displaying multivariate health care data. With respect to the literature, target plots are unable to display the minor gaps clearly between indicators, conversely radar plot is influenced immensely by displaying the order of the indicators. Nevertheless, Spie chart can successfully cope

[1] `http://office.microsoft.com/en-001/excel/`; visited on September 19th 2013.
[2] `http://office.microsoft.com/en-001/word/`; visited on September 19th 2013.
[3] `http://office.microsoft.com/en-001/powerpoint/`; visited on September 19th 2013.

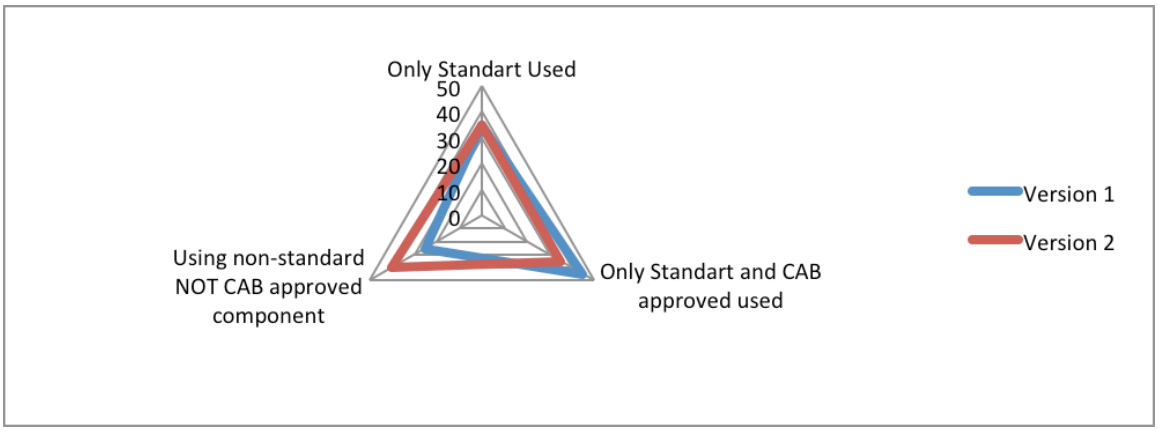


Figure 4.6.: Radar Chart.

with all of these restraints [SLF+11].

Requirement type - 6 (Table 3.6) addresses the security and the compliance properties of IT-products. Although this requirement aims initially to show limited properties of IT-Products, it can also be applied to express all the properties of IT-Products, technology or any kind of enterprise related to the product or item.

In order to address a solution for the requirement type - 6, the matrix representation is used. As in this requirement type, the mapping between the properties and the product should be clearly identified. Matrix representation offers a very compact view by explicit labelling property. Thanks to this property, instead of writing some letters to indicate the status of the properties (exp. Yes/No), elementary shapes (exp. Circle) can be used to indicate availability of the properties.

Figure 4.7 shows a sample of the matrix representation. When the matrix table fills in a lot because of the overloaded matching between the product and properties, the filtering mechanism can be applied. Owing to the filtering mechanism, the end user can filter the properties and express solely the chosen ones. This property might make the visualization more understandable and space efficient in the case of a big amount of product and properties.

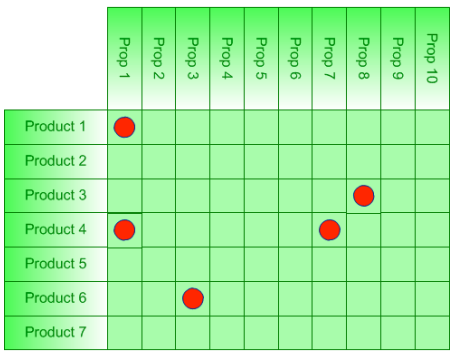


Figure 4.7.: Matrix representation only with one version.

Requirement type - 6 (Table 3.6) does not only aim to cover the correct mappings of products and properties, but also intends to compare different versions of them. Therefore, the matrix representation with different layers has proposed. For this purpose, color-coding has used to represent the different version. For example in the given example at Figure 4.8, the legend, which makes the visualization self-explanatory, explains the different versions by a different color expression.

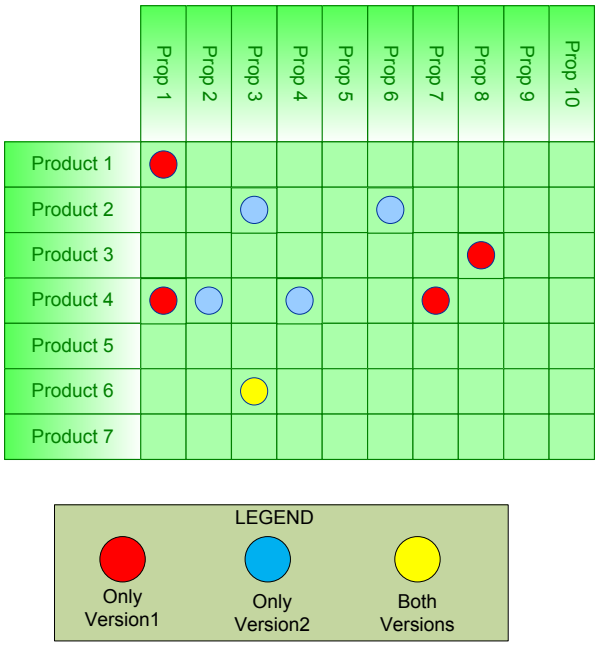


Figure 4.8.: Matrix representation with version comparison.

The visualization called Circos [Ngu12b] is a circular graph, which is used to visualize the relationship between the elements of IT Products and Systems. Originally, this innovative approach was created firstly to compare the similarities and differences between the genomes [Ngu12b], but it can be applied also to the other fields. This visualization can be created automatically with the whole software package by defining the inputs and configuration files [KSB+09]. In addition, apart from displaying the relations, it can specify the elements by scatter, line and histogram plots, heat maps, text etc. By using the online tool, Circos visualization is created at Figure 4.9 [ctv13]. To accomplish this, one sample data should be prepared in a tabular format and should be given as an input parameter. This visualization has prepared in order to address a solution for visualization of dependencies in EA Management. Circos can be a perfect example in solution for **Requirement type - 8** (Table 3.8). As it can be clearly seen in Figure 4.9, the dependencies between the different items can be obviously observed. Moreover, it can visualize the multiple dependencies and the percentage of them.

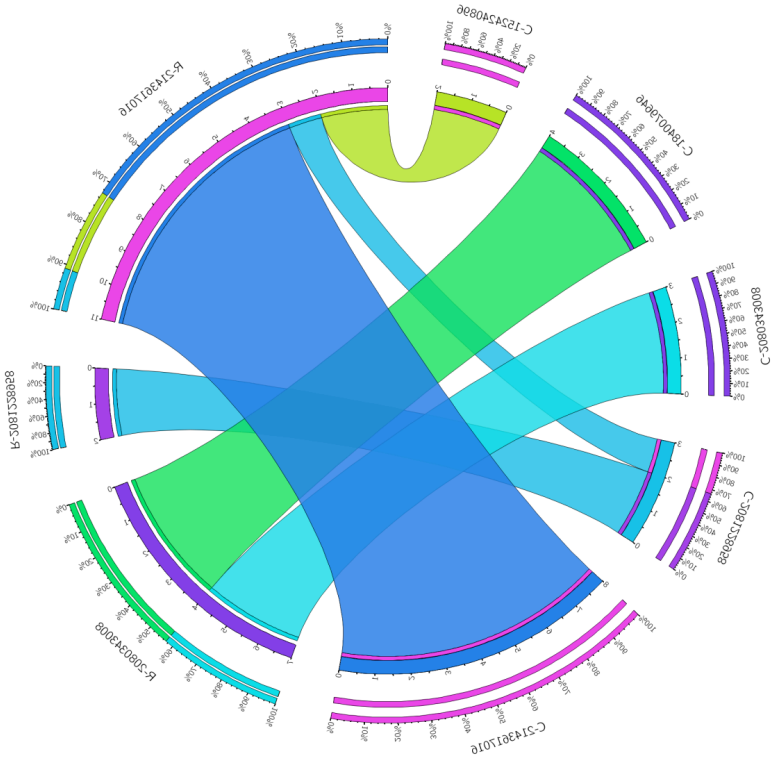


Figure 4.9.: Circos [ctv13].

The **requirement type - 10** (Table 3.10) and **requirement type - 11** (Table 3.11) are defined under the same solution. The reason is, the problems address by both requirements are very similar to each other and likely can be solved under the identical visualizations concepts. Both of these requirement types aim to find an answer to efficiently visualize the evaluation of the portfolios and observing the changes per portfolio during different states. Other than the overall change, these requirements are also aim to observe the changes or change of trends of the portfolio according to the particular semblance of them. To have a scenario for these requirements, the sample portfolio for IT-Product of one country can be assessed. To be more precise, with these requirements, the end user aims to observe the status changes of these IT-Products like *productive*, *planned* or *shutdown* ones in a particular country easily.

As stakeholder strongly mentioned, the visualizations should be easy and well-known ones, generally this concern considered for the visualizations. The reason for this, generally the employees are tending to handle this kind of visualizations because of high level of predilection. According to the stakeholder, generally the employees or decision makers do not want to spend time to understand the visualizations. In the light of this, the portfolio changes of one country should be observed by stacked bar chart which allows to contain several dimensions because of its layered structure (Figure 4.10).

Bar chart is highly used in the business because of its simplicity and easiness. However, when it comes to comparison of several attributes in different states, bar chart fails to do

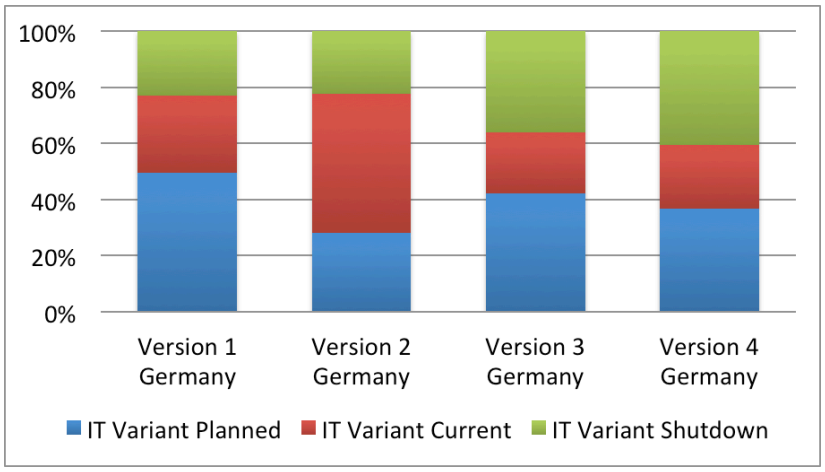


Figure 4.10.: Stacked Bar Chart - country based.

it because of its disability to show different layers. Stacked bar chart is intuitive visualization to show different layers of different attributes. Figure 4.10 shows, different states of whole IT Variant portfolio of one country (in this example Germany) can be easily observable. Moreover, by representing every layer with dissimilar colors, different states can be compared with each other with different versions and also can be compared with the other states.

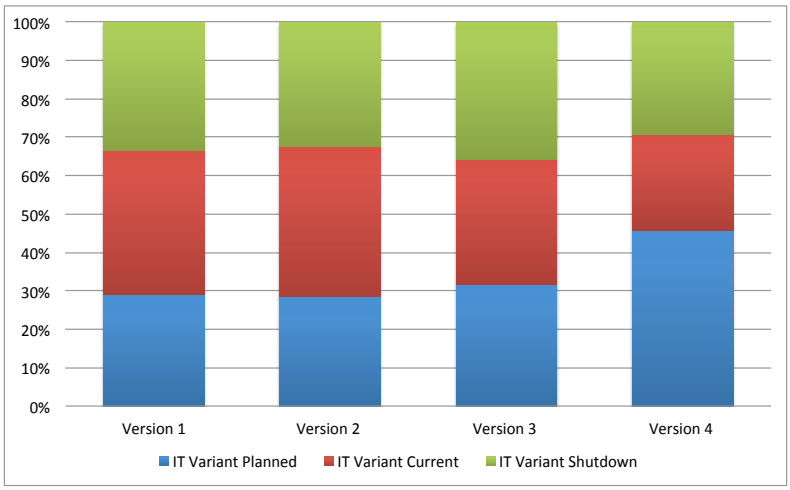


Figure 4.11.: Stacked Bar Chart - portfolio based.

Further, Figure 4.11 represents another variant of stacked bar chart. The reason is that end users often want to visualize the trend of changes in the whole portfolio. Thus, as a next step from this visualization, more specific or detailed information for each version may want to be visualized by the end users. In order to achieve this, the visualizations called cluster map can be used to specify the elements in the portfolio in more detail. Cluster map literally groups the elements according to their similarities. This visualization

offers a compact view to visualize the data that are in the same field. Figure 4.12 shows the representation of a country that is also visualized in the stacked bar chart. It shows the IT portfolio of one location according to the specified attributes. Cluster map can use color-coding in order to differ the attributes inside the cluster map. This helps the visualization to include another dimension. The cluster map may have space and scalability problems, in case it gets very populated. To cope with this problem, the layered cluster map is proposed, which provides more compact view by defining each layer for different attributes (Figure 4.13). End user can have a transition between the layers by click action. According to this action, the respective layer can be clicked, and the selected layer may come to the front. Layer approach may be automated by adding some animations other than the interactions. For instance as a concept, in every determined time interval one layer can come up to the front. Although this solution offers great usability opportunities, it may have some problems for the decision makers. Each user may have a different perception time for each layer, which makes this automated visualization inefficient in certain circumstances.

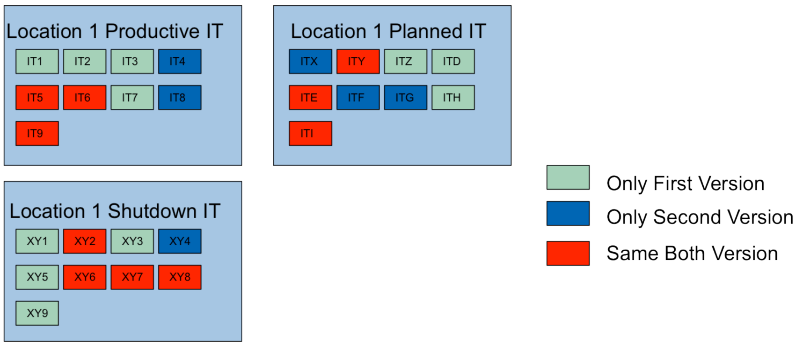


Figure 4.12.: Cluster Map with versions for specific location.

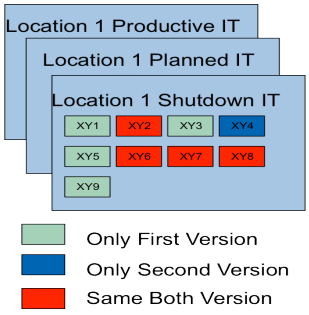


Figure 4.13.: Layered Cluster Map for specific location .

The cluster map can be broadened to visualize different kinds of data. Thus, it can be used to represent not only the country portfolio, but also the whole IT Product portfolio. At this point, different colors in the cluster map can be used to represent different attributes of the IT Products (Figure 4.14). Moreover as an improvement of this figure each cluster map can be shown in more compact view like displaying them in a layered format. In Figure 4.15 layers can be used to represent different versions or states of the whole portfolio, which allows users to compare different versions of the portfolio more efficiently. Figure 4.16 represents another type of visualization, which again allows users to compare efficiently the versions. Differently from the Figure 4.15, there is a timeline, which informs users about the available states or versions and enable more convenient control over the layers. It is less ambiguous and more user friendly. As it can be seen obviously in Figure 4.16 different colors assigns to the circles in order to represent different states of the versions. For instance, in the given example, red color indicates that there is a version at that date but not selected. Moreover, green color indicates the active version or state and white indicates the unavailability of the version at the given date.

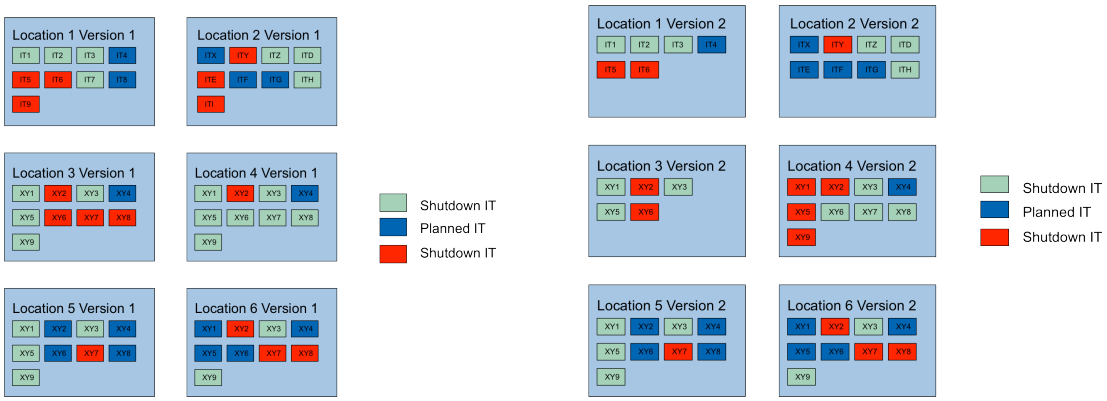


Figure 4.14.: Cluster Map representation of whole workspace.

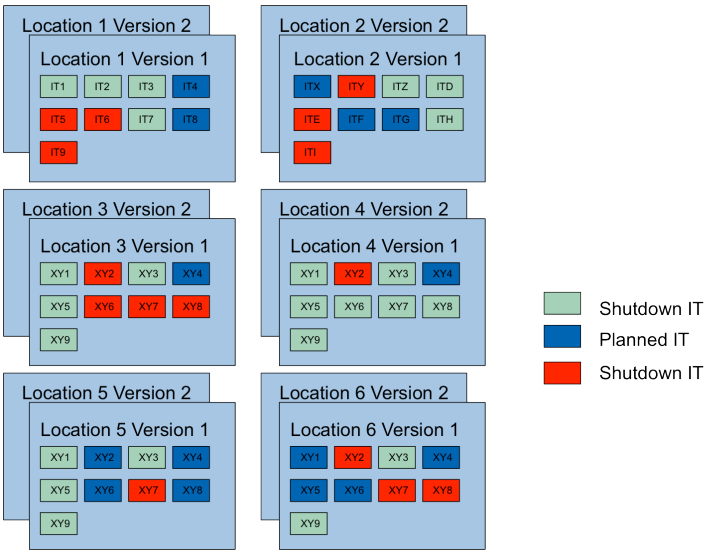


Figure 4.15.: Whole Space representation with Cluster Map.

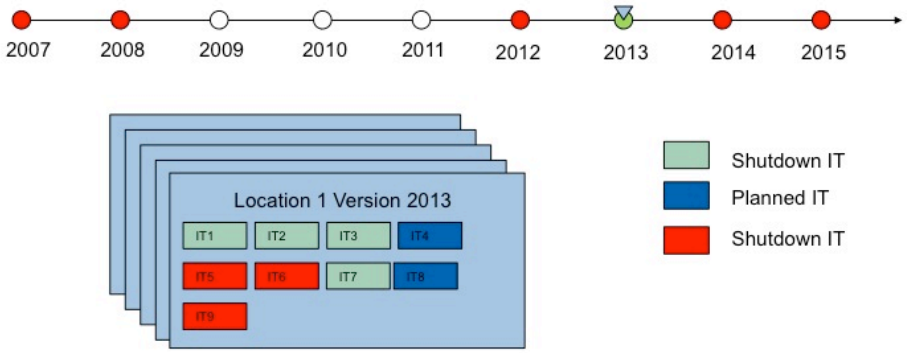


Figure 4.16.: Layered Cluster Map with timeline.

As a last step in conceptual visualization, Figure 4.17 proposes another variation of the visualization represented in Figure 4.16. Instead of representing the situation of the version in the time frame like in Figure 4.16, in Figure 4.17 the layers or cluster maps are directly represented in time scale.

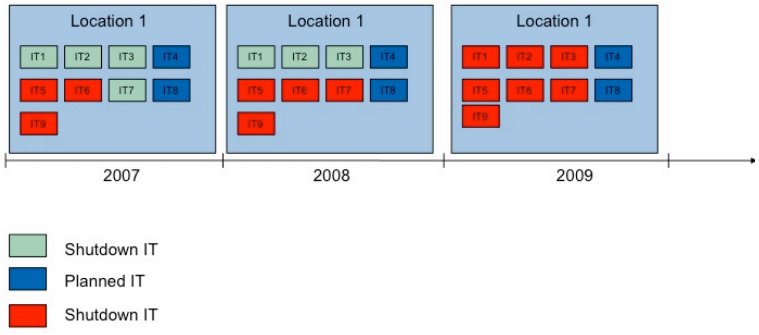


Figure 4.17.: Cluster Map with time frames.

The **requirement type - 9** (Table 3.9) looks for an answer for the location-based visualizations in the enterprises. It is basically seeking for a solution to visualize any item in the enterprise, based on the specific locations or countries. In order to achieve this, the map visualization will be formed that contains some other visualization like bar chart or cluster map. To be more precise, all of the implemented visualizations will be merged to the map by including some interactions like click etc.. In other words, the map of the enterprise will behave like a dashboard which contains all the relevant visualizations explained before in this chapter.

Figure 4.18 is the representation of a sample map. This map represents the particular locations or countries of an enterprise data. While representing the map of the enterprise, color-coding is used to show the active locations of the enterprise in the Europe map. In this visualization, the idea is to add some interactions to the map and show all the visualizations that were implemented before. In order to achieve this idea, the click interaction will be used which will allow the end user to interact with the active countries on the map (Figure 4.19). Moreover, thanks to this interaction the end user may choose active countries and select the visualizations that are assigned for that country(Figure 4.20). For each selected country, available visualization will be presented in a menu like visualization, which allows end user to choose their desired visualization. According to the concept, these inner visualizations would include all the visualizations that are country specific (Figure 4.10 to Figure 4.13).

Figure 4.18.: Sample Map.

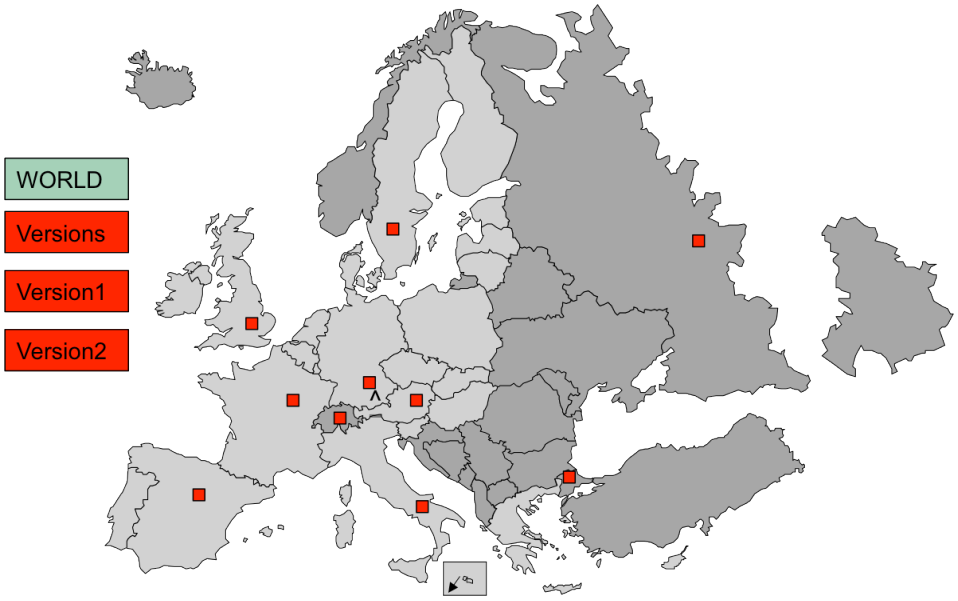


Figure 4.19.: Interactive map filled with data (a).

In addition to map visualization, this solution offers another option to visualize the whole workspace different from on the map. It allows the end user to view the whole portfolio in a plain format without including the other visualizations. The buttons on the left allows end user to choose from various visualizations in order to represent the whole portfolio which was represented before (Figure 4.21). This visualization is integrated to support the **requirement type - 10** and **11** which have mentioned before in this chapter.

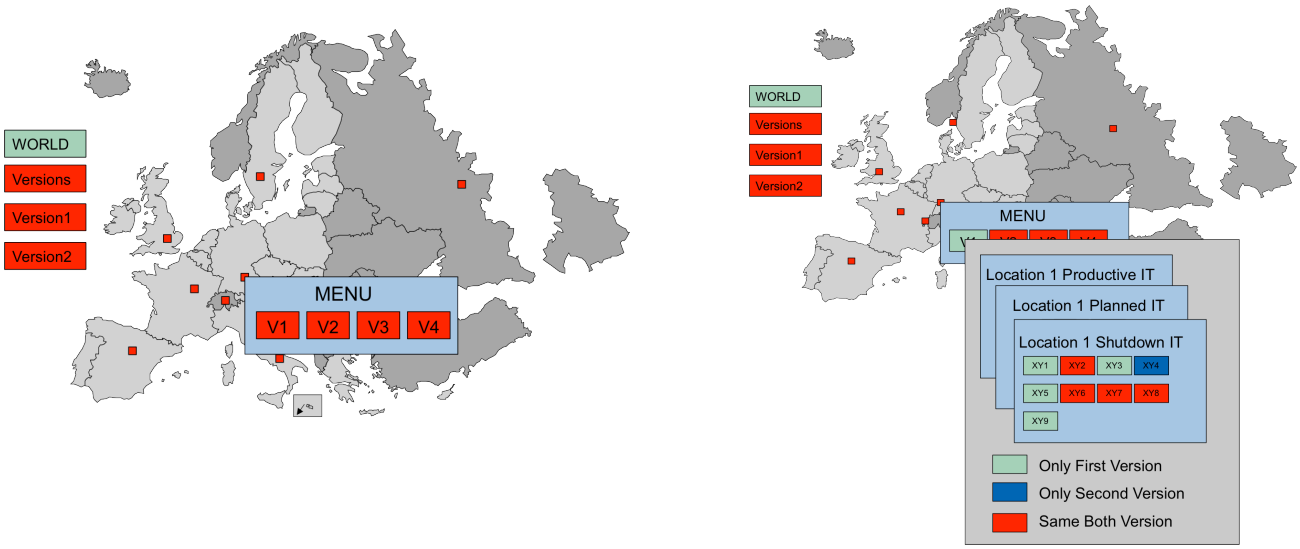


Figure 4.20.: Interactive map with Interaction (b).

To sum up, the map visualization offers compact visualization which summarizes all the implemented visualizations. The end user either can visualize information about the specific countries or the whole portfolio. It offers comprehensive solution for most of the requirement type offered by the stakeholder. Because of the time limitation of this research, not all the requirements will be implemented. Some of the requirements will stay as the conceptual work and candidate for the future work. As it is mentioned before the requirement types from 4-11 are designed for the visualizations. The visualizations of requirement type from 9-11 will be implemented. The visualization of other requirement types are only conceptualized and the implementation of them will stay as future work.

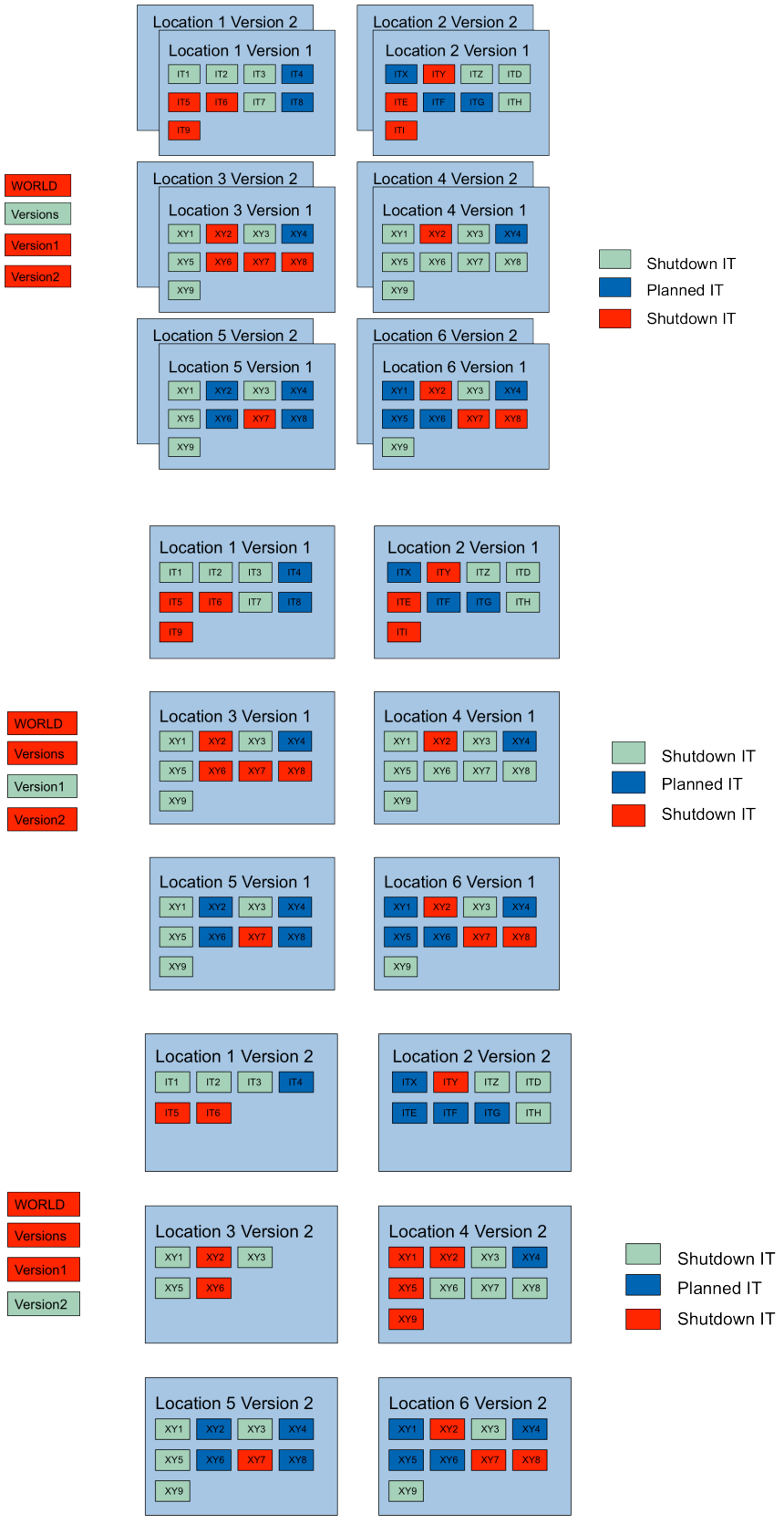


Figure 4.21.: Interactive map with cluster map view .

Part III.

Prototype Implementation

5. Prototype Implementation

The following chapter presents the detailed information about the implementation phase of the master thesis. First of all, it will introduce in detail the implementation scenario to explain the implementation steps. Then, according to the scenario, the import mechanism, version mechanism and visualizations will be explained in more detail.

5.1. Implementation Scenario

In EA Management research field, it is very important to combine theoretical and practical work. Therefore, it is often beneficial to realize and verify the theoretical work with industrial partners. Since this research held with one of the biggest consumer goods company in the world, there was a decent chance to assess it in real working environment.

Figure 5.1 shows general overview of the implementation scenario. This scenario will be applied throughout the implementation phase of the thesis. The scenario consists of four different parts.

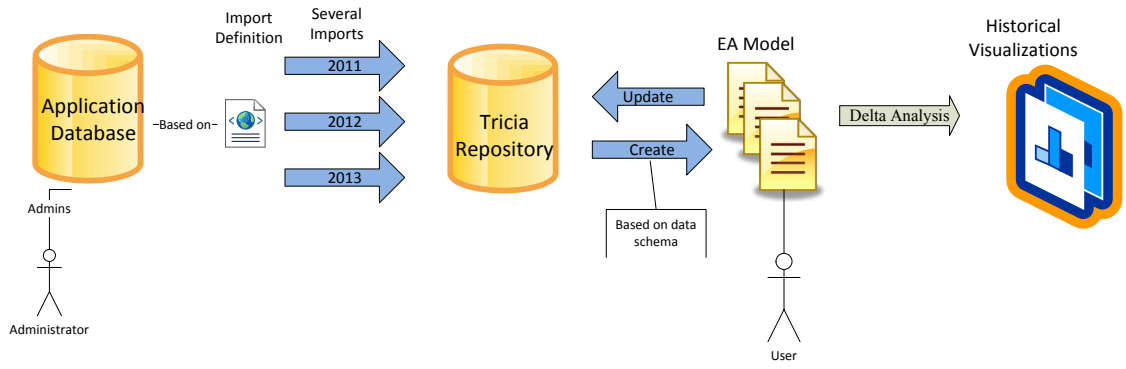


Figure 5.1.: General overview of the implementation scenario.

The first part starts with the preparation of the data schema and the data source. According to the logical model of the data source, the import mechanism of Tricia will be prepared. Thanks to the import mechanism, the enterprise data will be stored at internal data source of Tricia. As a second part, EA models can be created based on the data schema inside the Tricia internal EA repository. These EA models can be updated by the respective users of the systems and then the changes in EA models are stored back to the EA internal repository. This is the starting point for retrieving the different states of the EA models since inertial Tricia EA repository can hold the record of the changes. Applying delta analysis to the EA models among these records of the changes, different versions can be handled. Lastly, in the third step, the historical visualization will be created to visualize different EA models based on the retrieved versions. This aims to visualize the evaluation of EA models in an understandable and logical way. These historical visualizations are conceptualized in Chapter 4.

5.2. Data Model

As a preparation for the implementation phase, sample data model is prepared based on the stakeholder's enterprise data model(Figure 5.2). Conceptually, the whole IT portfolio of the stakeholder is desired to be reported in the Tricia.

According to the data model, the main object is IT Product. IT Product consists of several variants called Variant IT Product. Each IT Variant maybe available in several countries, locations or regions. Moreover, Each IT Variant has several objects namely; process, information object, technological component and system component. These concepts are similar in all consumer goods industry in the world. Thus, in this research, these concepts will not be declared in more detail.

As it is mentioned before, Tricia consists of several Space objects, that contains PageSpace and UserSpace objects. PageSpace object contains a list of the pages with the same type of the PageSpace object. These concepts are the main issues in Tricia to display the data correctly. Thus, it is necessary to have a correct logical mapping between the data model of Tricia and enterprise information model. Figure 5.3 illustrates an example where there is a PageSpace called Geographical Information. Geographical Information PageSpace consists of several set of pages. To be more precise, Geographical Information space includes the set of pages in Country type that contains a list of Country pages.

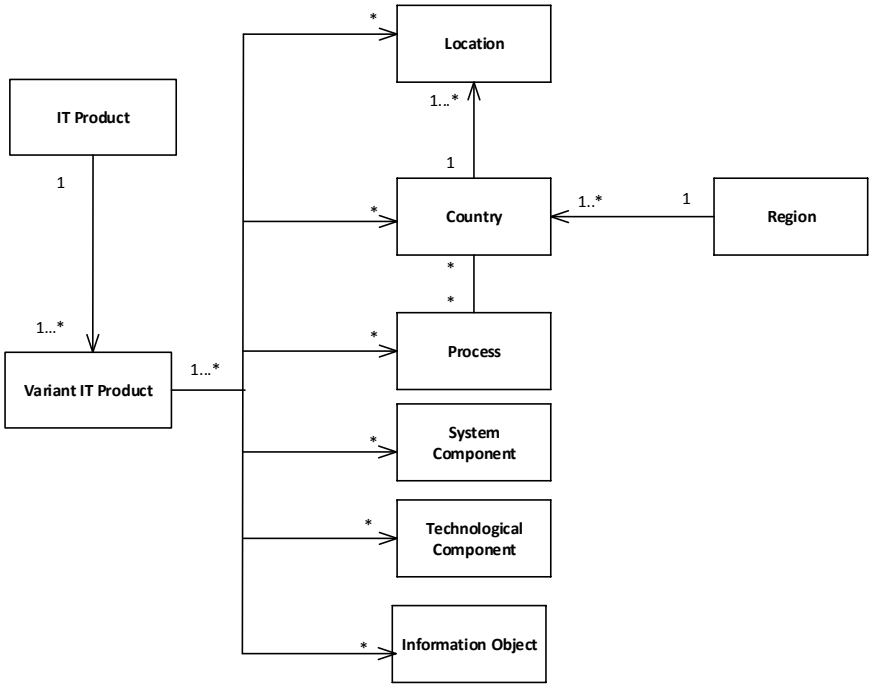


Figure 5.2.: Logical data model for the scenario

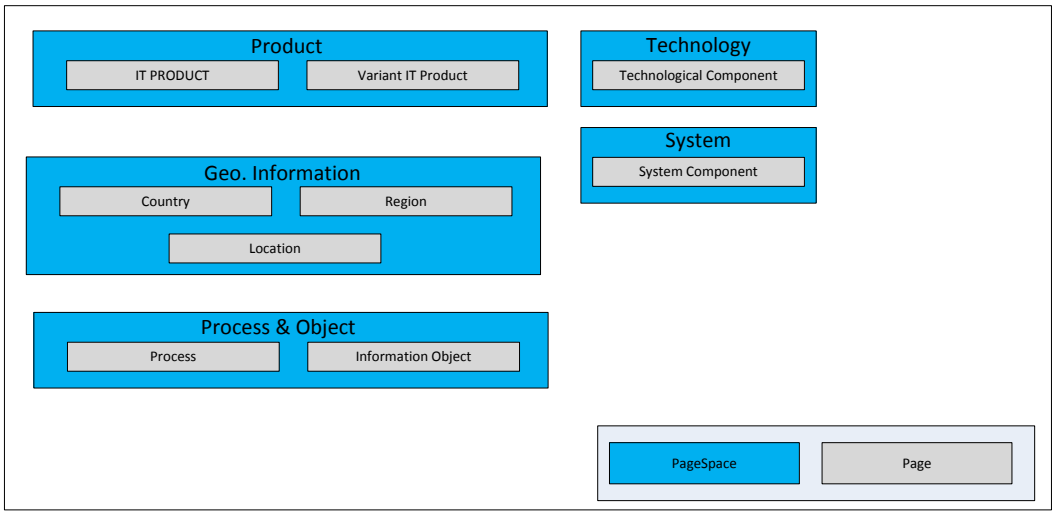


Figure 5.3.: Mapping of PageSpace and Page

5.3. Import Mechanism

Tricia allows users to import data automatically to its internal repository. Automatic import mechanism in Tricia has been improved to import huge amount of data in a small amount of time. Especially when there are big changes in the data source, using an automatic import mechanism may save lots of time. From outer data sources like relational databases, spreadsheets and XML files, the data can be imported to the Tricia. Likewise, in Tricia data can be exported in spreadsheet format. Currently, Tricia may have different persistence database configurations like MySQL [1], Oracle [ORA13] and HSQLDB [2] for the testing purposes.

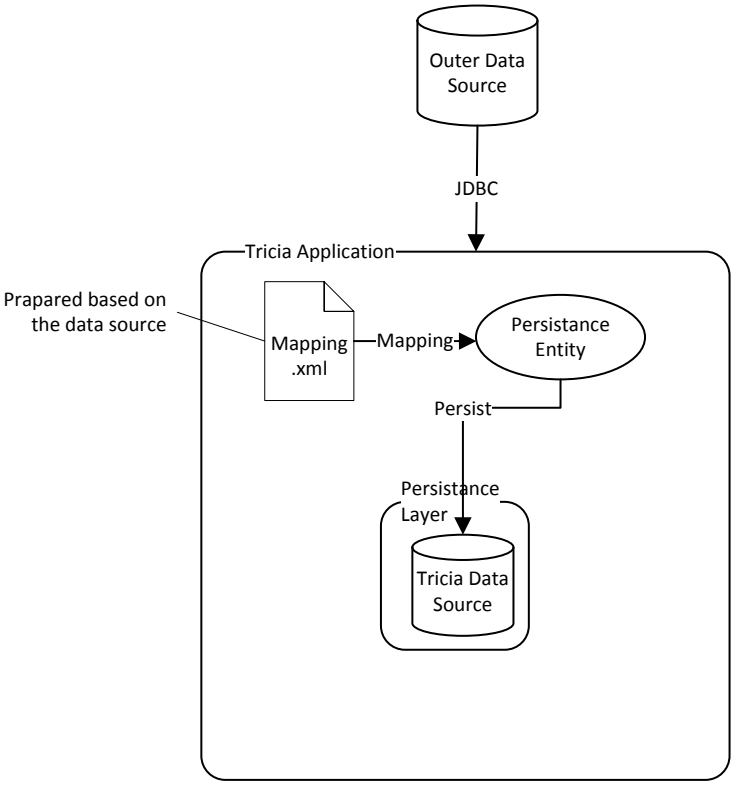


Figure 5.4.: Import mechanism

Figure 5.4 is a representation of import mechanism in Tricia. The outer data source is connected to Tricia framework by having *Java Database Connectivity*(JDBC) [3]. By this, Tricia framework can get any data from the data source based on the definition in the mapping file(Mapping.xml). On the other hand, currently the Tricia framework does not support any write-back operation. This disables Tricia to store back any changes to the outer data source. After the correct connectivity, the data queried from the outer data source to the

[1] `http://www.mysql.com/`; visited on September 23th 2013.
[2] `http://hsqldb.org/`; visited on September 23th 2013.
[3] `http://www.oracle.com/technetwork/java/javase/jdbc/index.html`; visited on September 23th 2013.

Tricia. In order to have this, *Extensible Markup Language* (XML) [4] file is prepared based on the outer data source schema. According to the information written inside the tags of the XML file, the parser queries the data from the outer source by creating the predefined queries. The PersistenceEntity superclass mapped by the defined queries. After assigning new values to the entities, they map to the persistence layer of the Tricia (Figure 5.5).

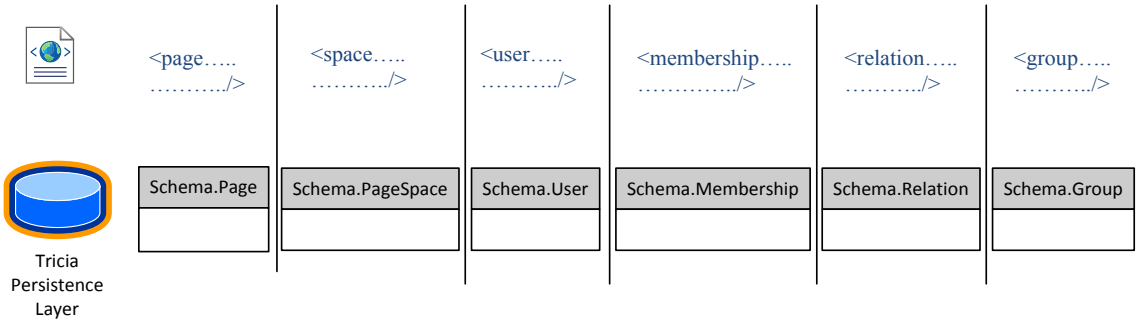


Figure 5.5.: Mapping of XML to Tricia repository.

In the following paragraphs, the import mechanism will be explained in more detail by giving examples from the mapping file. However, this explanation will be based on some examples and will not cover all the elements of the particular tags. In order to investigate the whole elements and the structure of the XML document, the schema of the document should be glanced Appendix A.

The first step for defining the import XML file should be to introduce the relational database connection. During the implementation phase of this research, MySQL and Microsoft SQL database connections are used. Moreover, any other database connection can be supported when the related connection provider is implemented, and libraries are imported in the local server. In order to define databases conveniently, the XML elements like, JDBC , location of the data source, username, and password of the database should be correctly defined. Also, one suitable Uri from the local server should be provided, which will be created automatically by the system to store the transferred data inside the server. To have a successful database connection, JDBC connection should be provided. In this case, the related libraries should be imported either for Microsoft SQL Server(MSSQL) [5] or MySQL. Tricia import mechanism has a capability to import files for the pages. File is imported in order to support the content of the pages. The mapping tag in import mechanism has *documentstoreroot* node which defines the directory that has list of all the files(Figure 5.6).

[4] `http://www.w3.org/XML/`; visited on September 23th 2013.
[5] `http://www.microsoft.com/en-us/sqlserver/default.aspx`; visited on September 23th 2013.

MSSQL connection

```
<mapping connectionprovider="com.microsoft.sqlserver.jdbc.SQLServerDriver"
       connectionurl="jdbc:sqlserver://localhost;databaseName=import;user=root;password=root"
       connectionusername="" connectionpassword=""
       documentstoreroot="C:\production\documents”>
```

MySQL Connection

```
<mapping connectionprovider="com.mysql.jdbc.Driver"
       connectionurl="jdbc:mysql://localhost/import" connectionusername="root"
       connectionpassword="root" documentstoreroot="C:\production\documents">
```

Figure 5.6.: Connection to database

Tricia enables enterprises to have a collaborative working environment. As a consequence of this, in order to reduce the complexity and to enable less chaotic collaborative work, Tricia has the functionality to store each user and ability to group them. This is the initial step of the successful access control in Tricia. Further, this allows to track user changes, and it provides the customization of EA system per user [Neu12]. Figure 5.7 gives an example of how to introduce new groups, users and memberships. For all elements, the name of the table should be defined as an attribute. Afterwards, related typing can retrieve data from the outer database table and maps fields in the Tricia. Moreover, the user can filter the results by introducing *filter* keyword, to restrict or filter the results without creating any extra keywords.

```
<group tablename="(SELECT 'Editors' as name, 'editorGroup' AS id) AS embedded"
              idfield="embedded.id" namefield="embedded.name" />
       <user tablename="employer" loginfield="uemail" passwordfield="upassword"
              idfield="xname" namefield="CONCAT(uLastName,', ',ufirstName)">
              <fields>
                     <field columnname="uFirstName" tricianame="First Name" triciatype="Text" />
                     <field columnname="uLastName" tricianame="Last Name" triciatype="Text" />
                     <field columnname="uDepartmentName" tricianame="Department Name"
                            triciatype="Text" />
              </fields>
       </user>
       <membership tablename="employer" groupfield="'Editors'"
              groupfieldtype="name" userfield="uemail" userfieldtype="name" />
```

Figure 5.7.: Import group, users and membership

The Tricia tool has a powerful concept that can group the content under several dimensions like in the space, and in the page. Thanks to the import mechanism, these concepts can be easily arranged and mapped with the data source. To define a space, only the name of the space should be given (Figure 5.8). In addition to that, whole space accessibility restrictions can be arranged while defining the space. For instance, the readers and writers of the space can be restricted by groups. The memberships and groups are defined before in the XML file by the nodes called *membership* and *group*.

As it is mentioned before, each space consists of several PageSpace and pages. In order to define the PageSpace under the predefined space, page node should be filled. Inside the page node, different type of page instances can be defined. The element *tricianame* is the name of the page type; *tablename* is the name of the table in the outer data source to

retrieve pages for the defined type. The different page types or subspaces behave like a page container. Elements that are called *namefield* and *idfield* are used to identify the unique wiki pages. As an example, space called "Geographical Information" has a subspace called "Country" with the same type list of pages which gets the pages from the outer relational database table called "V_Tricia_Entitiy_25_Country". From the given relational database table, each page is mapped to the column called "Country" and "CountryCode". Moreover, there is an active access control in wiki pages similar in the space. Access control of the pages can be defined by setting the readers and writers group (Figure 5.8). Predefined groups can be assigned to read and write roles.

The structured part of the page is filled by the element called fields. As it is stated before in this work, structured content or hybrid consists of attribute - value tuples. Inside the field element, the attribute is defined using the element name called *tricianame* and it is mapped to the column of the relational database table, which is defined with the element name called *columnname*. The value of the defined attribute is retrieved from the relational table. Besides these functionalities, the type of the tuples can be determined with the element called *triciatype*. The type of the tuples can vary from Text objects to Link objects considering the needs of the importer.

```
<spaces>
	<!-- Page 1 -->
	<space name="Geographical Information">
		<readers>
			<reader type="system" value="registered" />
		</readers>
		<writers>
			<writer type="groupname" value="Editors" />
		</writers>
		<pages>
			<page tricianame="Country" tablename="[V_Tricia_Entity_25_Country]"
				namefield="[Country]" idfield="[CountryCode]">
				<fields>
					<field columnname="[CountryCode]" tricianame="Country Code"
						triciatype="Text" />
					<field columnname="[Country]" tricianame="Country"
						triciatype="Text" />
					<field columnname="[Area]" tricianame="Area"
						triciatype="Text" />
				</fields>
			</page>
		</pages>
	</space>
```

Figure 5.8.: Import spaces and pages with access control

Generally, mid-sized and big companies have a complex, intertwined and highly coupled IT infrastructure. Thus, it is necessary for stakeholders to observe relations in order to show the connections and dependencies inside the company and predict analysis about their company. It is often important for decision makers to see these relations and make analysis on these relations. Consequently, it is an indispensable feature for EA Management tools in order to indicate the relations clearly between different attributes or elements.

After creating the spaces and pages in Tricia, the relations between these pages should be created. The relations should be defined after importing the pages and spaces because

stating a relation between pages that are not created before in Tricia internal repository is logically impossible.

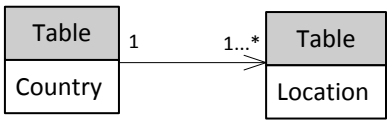
In Tricia, one to one, one to many and many to many relations can be defined. To define different relations, user must identify different types of queries, which are same as SQL queries. In the scenario data, SQL views are used directly to get rid of the complexity. However, users can write *join* statements instead of defining the SQL views to define many to many relations.

To create any type of relation, *tablename*, *fromkeycolumnname*, *tokeycolumnname* and *filter* should be diagnosed in *relation* tag (Figure 5.9). These elements are used to create an SQL query inside Tricia. Other elements like *tricianame*, *fromparent* are used to organize appearance of the relation inside the Tricia system. For instance, *tricianame* shows the relation name that should be represented in the system. *fromparent* and *toparent* explain mapping mechanism of the result set retrieved from the query. In order to create the relations, the queries similar to the following sample are produced:

- "**SELECT** *fromkeycolumname, tokeycolumnname* **FROM** *tablename* **WHERE** *filter*".

For many-to-many relations, join statements are defined. This join query should be written inside the *tablename*. In other words, Tricia system is not automatically detecting the many to many relations. Thus, it should be defined inside the XML mapping file.

1-1 & 1-N Relation



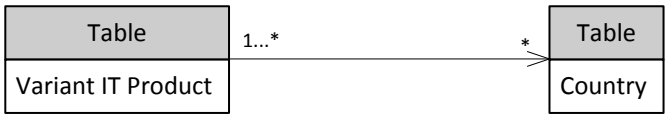

```
<relations>

        <!-- Relation Country to Location -->

        <relation tricianame="Location"
                tablename="[V_Tricia_Relation_25_CountryLocationMapping]"
                fromkeycolumnname="[CountryCode]" fromparent="Country"
                tokeycolumnname="[LocationCode]" toparent="Location" />

</relations>
```

N-N Relation




```
<relations>

        <!-- Relation IT Variant to Country -->

        <relation tricianame="IT Variant Planned Next Year"
                tablename="[V_Tricia_Relation_25_Variant_to_Country] where
               [ID_ProductionStateValue]=2 and [Production_State_Year]=YEAR(getDate()+1)"
                fromkeycolumnname="[ID_Variant]" fromparent="Atlas Product Variant"
                tokeycolumnname="[CountryCode]" toparent="Country" />
</relations>
```

```
<relations>

        <!-- Relation IT Variant to Country -->

        <relation tricianame="IT Variant Planned Next Year"
                tablename="[V_Tricia_Relation_25_Variant_to_Country]"
                fromkeycolumnname="[ID_Variant]" fromparent="Atlas Product Variant"
                tokeycolumnname="[CountryCode]" toparent="Country"
                filter="[ID_ProductionStateValue]=2 and[Production_State_Year]=YEAR(getDate()+1)"

        />
</relations>
```

Figure 5.9.: Importing different relation types in Tricia

According to the scenario, each relation passes attributes to explain the relation in more detail (Figure 5.10). It is very important for decision makers of the stakeholder to see these attributes in the relations. According to them, plain relation representation does not have any sense and meaning. However, the Tricia import mechanism does not support this feature. For this problem, the workaround developed which aims to create different relations for each attribute (Figure 5.11). In order to accomplish the workaround, different SQL views created which contains the aggregation of relations of two different pages. These views contain only the id of the pages and the relation names between them, which have the attribute inside. At last, for each different attribute name, different relations are defined in the import file (Figure 5.12).

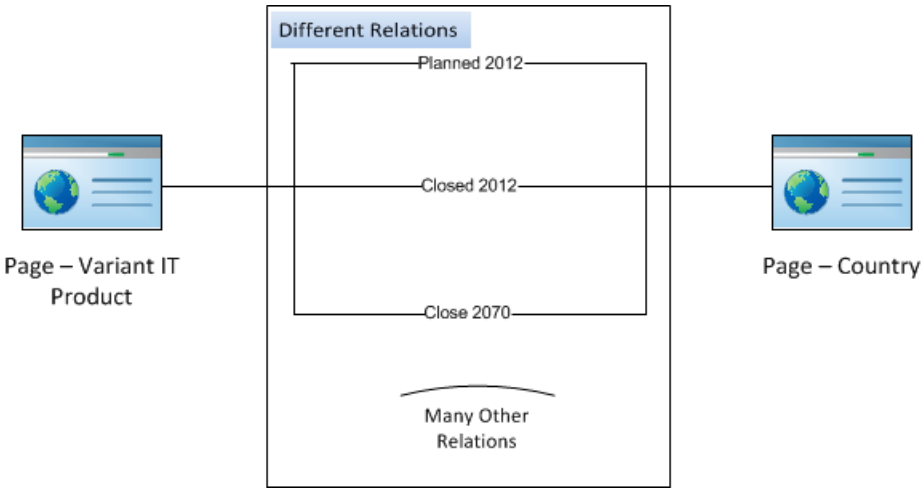


Figure 5.10.: Scenario data attribute passing problem.

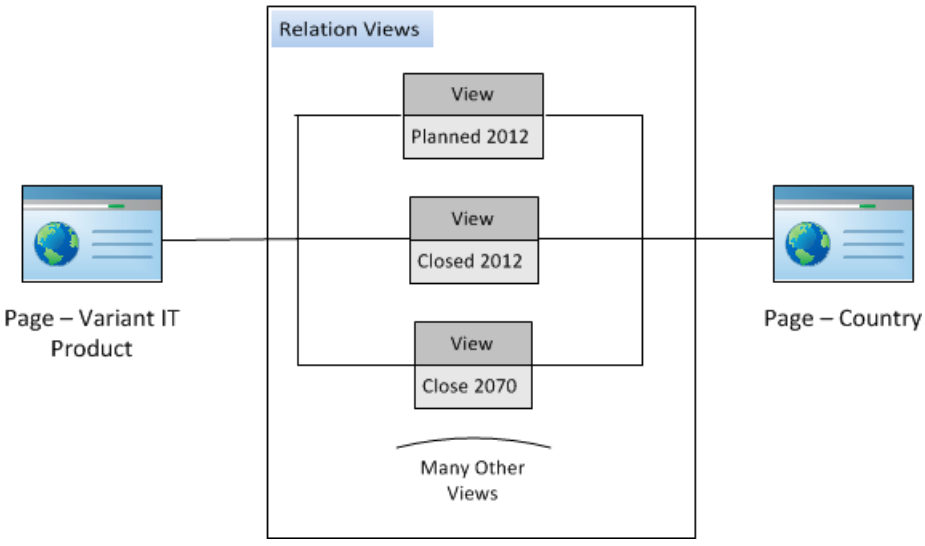


Figure 5.11.: Workaround for the attribute-passing problem.

```
<relations>

        <!-- Relation IT Variant to Country -->

        <relation tricianame="IT Variant Planned Next Year"
                tablename="[V_Tricia_Relation_25_Variant_to_Country] where
               [ID_ProductionStateValue]=2 and [Production_State_Year]=YEAR(getDate()+1)"
                fromkeycolumnname="[CountryCode]" fromparent="Country"
                tokeycolumnname="[ID_Variant]" toparent="Atlas Product Variant" />

</relations>
```

Figure 5.12.: Relation with attributes passing

Tricia system uses Web2.0, which gives many opportunities to the system as a reporting tool. With the improvement of Web2.0, a new concept called tag clouds or tagging has emerged. To create tags, the user either can define them on the Tricia itself or tags can be defined in the import mechanism. To define the tags, the importer can define another table or can use the fields from the already used tables for other types of mappings. As it is stated before, there are two types of tags in Tricia called Type Tags or standard tag. Currently by the import mechanism, the user only can import the normal tags. As a future implementation steps, the import mechanism can be extended to import Type Tags other than standard tag.

Tricia import mechanism first creates an entity and then persist it to its own persistence layer. When importer first creates the entity, it is in the ”write” state. This means that any update on the entity can be done freely. However, after the persisting the entity, the state becomes read-only. This disables users to persist identical entities several times. Thus, in the context of thesis work, import mechanism of Tricia is extended in such a way that it can update the data inside the persistence database in import mechanism. To deal with this limitation, the updates can be applied to the writeable copy of the entity. In other words, to Figure 5.13 resembles a small roadmap of updating the pages by using the import mechanism. Although the example is only for page, this property can be applied to all others entities (Space, Page, Tags, Relations etc.).

The user of the system first starts with **SqlImportProcess** after uploading the XML import file to the Tricia framework. Then according to each node and attribute defined in the import file the process calls the related functions to transfer data to the Tricia. When creating a page, there should be a control whether the page has existed in the system before or not. If it is not existed, then the system easily inserts a newly created page with the defined id. However, if given id is overlapping with the other ids then the system is creating the writeable copy of the page and works on that entity and updates the changes to the persistence schema.

While updating the page, naturally the hybrid of the page will be also updated. In order to update the hybrid of the page, the Versionable and Hybrid mixin should be used. Versionable mixin allowing Tricia to track the changes and Hybrid mixin is calling the Hybrid object of the page (Figure 5.14).

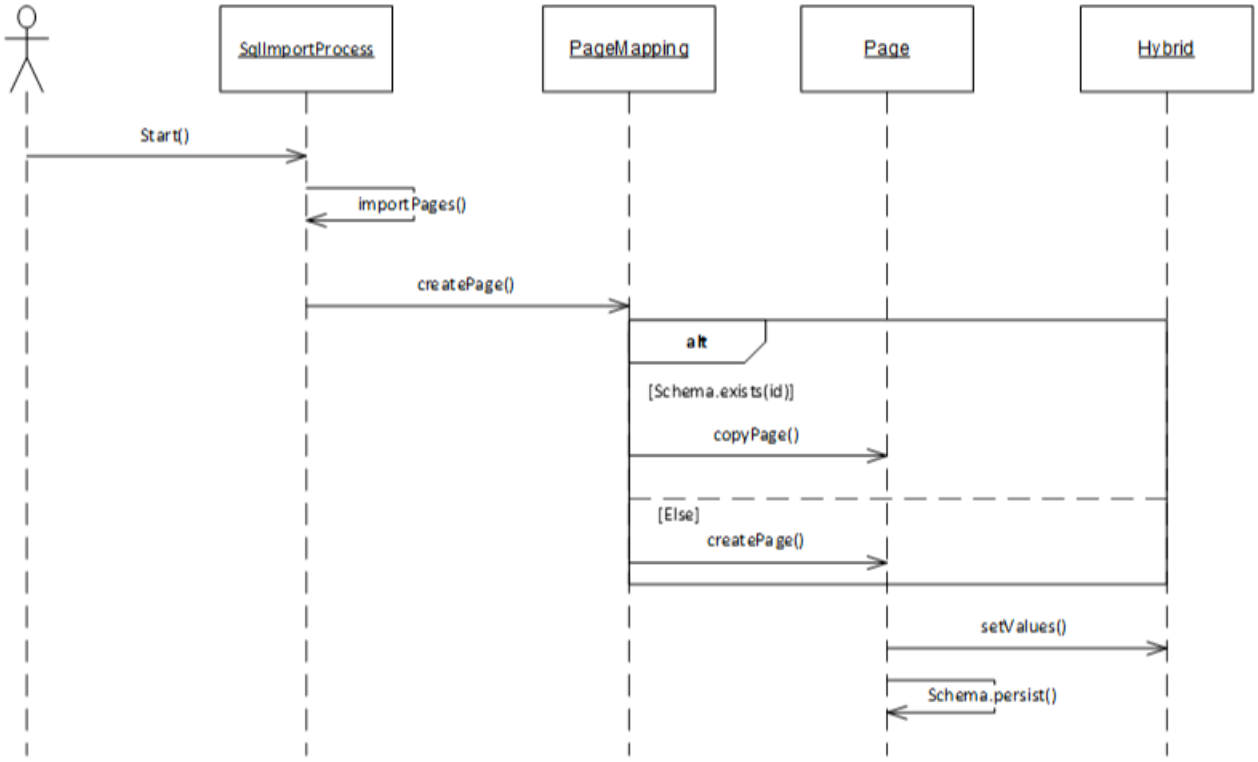


Figure 5.13.: Persisting entities constantly.

```
Page page = null;
Hybrid hybrid;
if(Page.SCHEMA.exists(id)){
      page = Page.SCHEMA.getEntity(id).createWritableCopy();
      hybrid = page.adapt(Versionable.class).adapt(Hybrid.class);
}else {
      page = Page.SCHEMA.createWritableEntityWithId(id);
      hybrid = page.adapt(Hybrid.class);
}
page.persist();
```

Figure 5.14.: Code sample for persisting different pages

5.4. Version Retrieving In Tricia (First Step for Historical Visualizations)

After preparing the enterprise data and having correct mapping and preparation of the import file, the Tricia system becomes ready for the imports. According to the implementation scenario, after the first import of the enterprise data from outer data source, the users may make changes on the EA instances or can continuously import outer data source and update the internal EA model repository (Figure 5.15). According to the figure, there are constant changes to the EA model by the users and additionally, in every year there is an update by the administrator withal using the import mechanism.

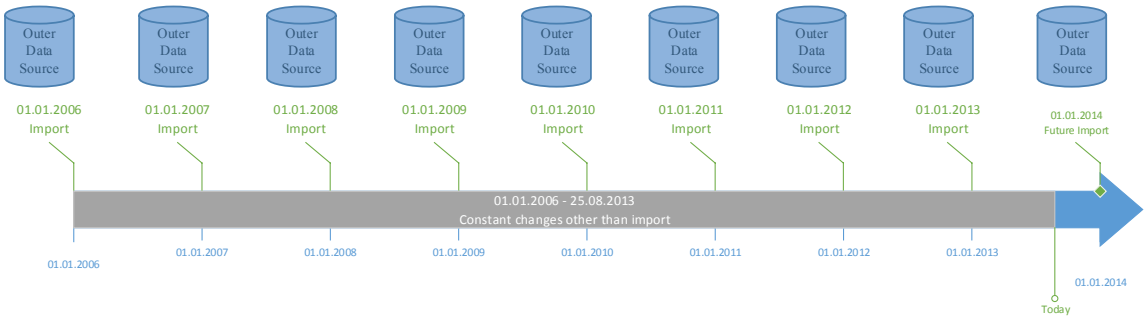


Figure 5.15.: Importing several versions.

Tricia system has mechanism to detect the changes made on the particular entities. In other words, Tricia has the mechanism to store each change in its internal data source. When there is a change in the system, Tricia stores these changes together with the owner of the action, timestamp and explanation of the change. To accomplish this, each page has its own ChangeSet object, which contains set of changes including the knowledge of timestamp. Timestamp information gives the system great flexibility for retrieving the versions of a specific date. As it is introduced before, Tricia consists of several spaces, which have several pages inside. Every page has at least one ChangeSet. Even Though, there are not any changes after the first creation of the particular page, the system is creating ChangeSet with the definition of the new page (Figure 5.16).

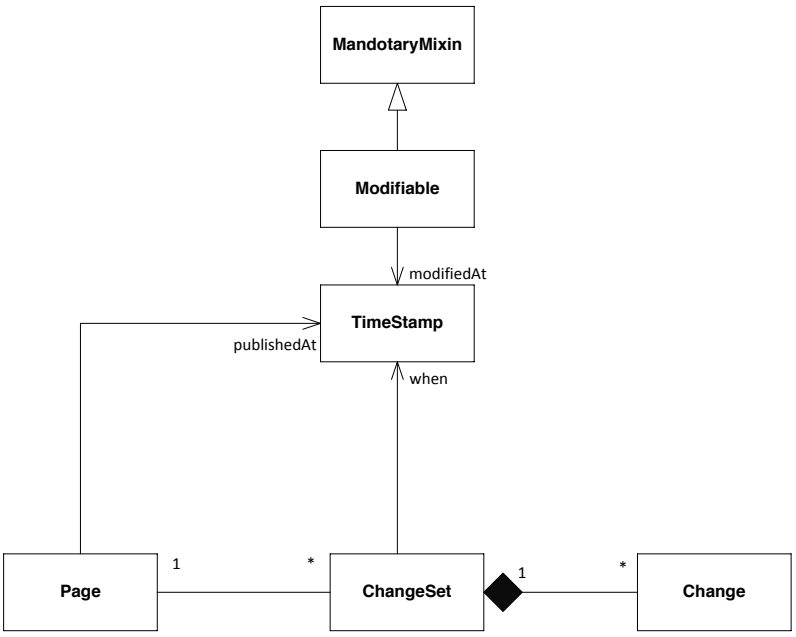


Figure 5.16.: ChangeSet class diagram .

In order to retrieve each version, Tricia system is extended in a way that, it can retrieve

the data from Tricia repository for historical visualizations. The first step of defining historical visualization is forming of the versions. As it is stated before, owing to awareness system in Tricia, all changes can be reached from the Tricia internal database. In order to attain particular versions, the changes are subtracted from the last retrieved version (Figure 5.17). For instance, according to the following figure, to retrieve the desired versions changes until the date of the desired version should be queried ChangeSet schema by using the correct TimeStamps. Afterwards, these changes are subtracted from the today's version of the Tricia to get the desired version.

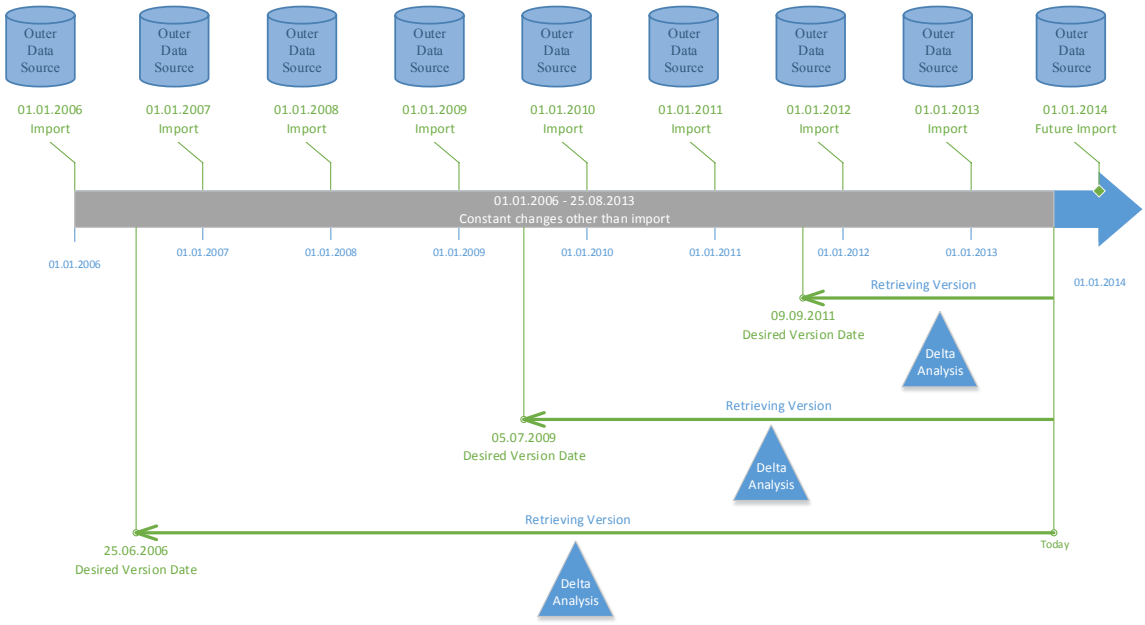


Figure 5.17.: Retrieving several versions.

Figure 5.18 is a sequence diagram of version retrieving for the particular page. First of all the user or the administrator of the Tricia should determine the desired date of the version. Then, these dates are set to the VersionHelper class, which handles all the version mechanism. Thereafter, retrieving from the page schema should create the requested page. Then, Query object is created, which is the fundamental object for retrieving particular versions. Query object creates an object to retrieve changes from the any schema in Tricia repository. The created query object queried inside the ChangeSet schema to get all the ChangeSet. After that all the changes inside the ChangeSet are iterated to update the page and thereof the hybrid of the page.

As it is mentioned before, Tricia has the ability to query its internal data source just like in the relational database. This gives great flexibility and speed to the programmer when retrieving specific information inside Tricia.

Query object in Tricia enables programmer to reach specific object and specific change set for particular timestamps. Query object is used as a query API to get the all the matching queries from the PersistentSchema (Figure 5.5). In order to retrieve the correct space or page, first of all, the correct version should be created. The following code snippet shows a way how to build up a correct query object 5.19.

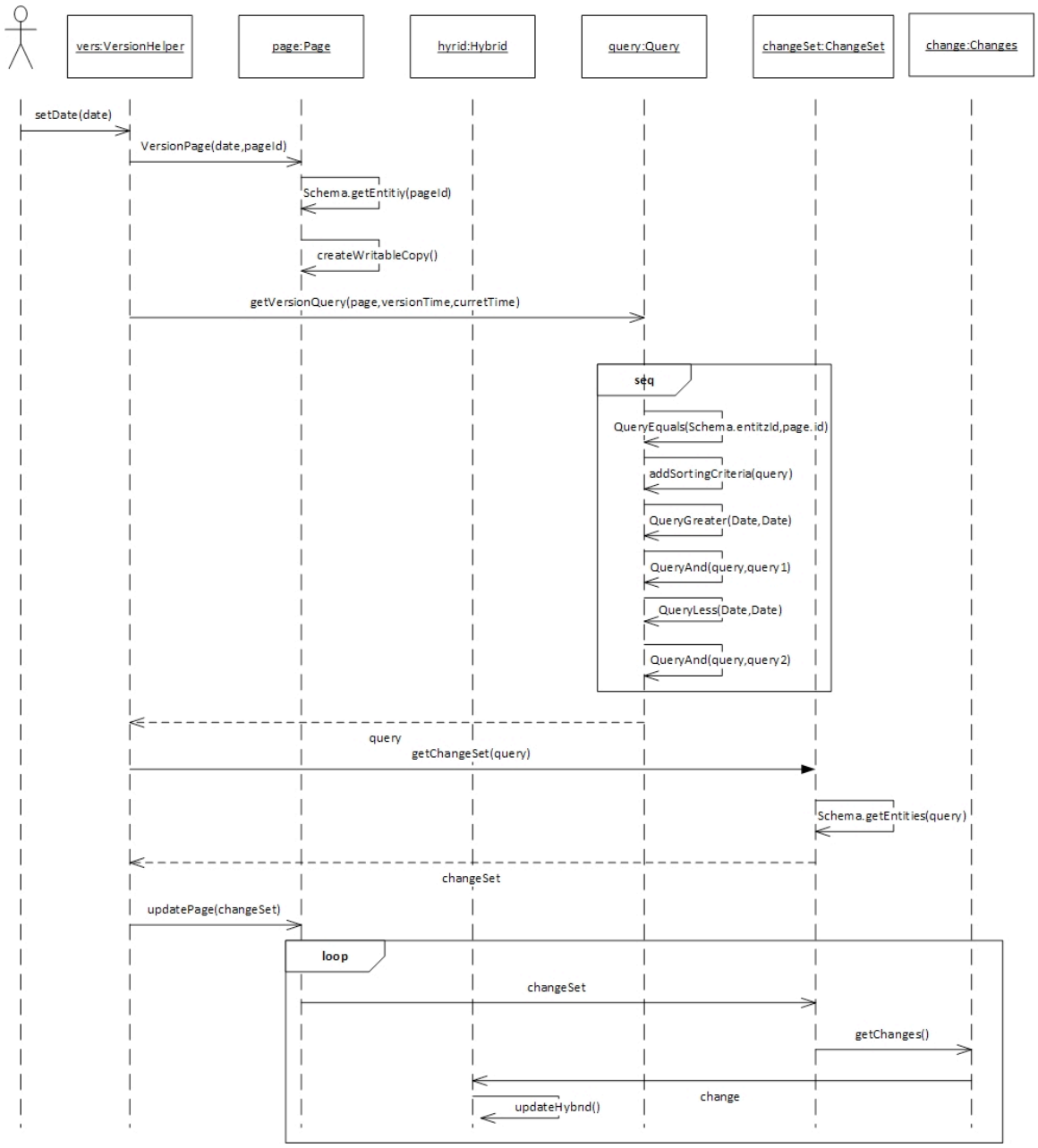


Figure 5.18.: Sequence diagram for version retrieving.

According to the code snippet, the function **getVersionQuery** produces a Query object for the given page and time. The function takes page, date for the version and the current time as parameter. The function first calls **QueryEquals** object, which basically compares and finds the given element in the schema type of the given object. For instance, in the given example, the function produces a query that matches the id of the changes with the id of the given page. If the result is wanted to be in a descending or an ascending order, the query should call **addSortingCriterion** method. **QueryGreater** and **QueryLess** objects

```
private static Query getVersionQuery(Page page, final String versionTime, final String aVersionTime) {

    Query query = new QueryEquals(((ChangeSet) ChangeSet.SCHEMA.prototype()).entityUid, page.getUid());
    query.addSortingCriterion(new Ascending(ChangeSet.SCHEMA.prototype().when));

    try {
        query = new QueryAnd(query, new QueryGreater(((ChangeSet) ChangeSet.SCHEMA.prototype()).when, new Timestamp(new
                SimpleDateFormat("yyyy-MM-dd HH:mm:ss", Locale.US).parse(versionTime).getTime())));
    } catch (ParseException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
    try {
        query = new QueryAnd(query, new QueryLess(((ChangeSet) ChangeSet.SCHEMA.prototype()).when, new Timestamp(new
                SimpleDateFormat ("yyyy-MM-dd HH:mm:ss", Locale.US). parse(aVersionTime).getTime())));
    } catch (ParseException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
    return query;
}
```

Figure 5.19.: Generating a query

enable Tricia to query the ChangeSet schema with timestamp information. For instance, in the given code, QueryGreater retrieves data that is greater than the given date. On the other hand, QueryLess retrieves data that is less than the given date. **QueryAnd** object is dealing with this problem by filtering two different queries. It is often important to formulate join statements in the queries.

5.5. Visualization Implementation

First of all, this subchapter will give a general overview about the visualization concept in Tricia more detailed by including the generic class diagram. Then the visualizations that are implemented in this research will be discussed in details.

5.5.1. General Overview

Tricia has a very strong conceptual framework for creating visualizations. Independently from the information model, Tricia can automatically produce business visualizations. This chapter will discuss, how the implementation of the visualizations can be realized by using this framework. As it is mentioned before, hybrid wikis are bringing the power of traditional wiki and semantic wikis together. In other words, hybrid wikis are supporting both unstructured and structured information. The unstructured data may include any text, picture, hyperlink, video or visualizations. In Tricia, thanks to what-you-see-is-what-you-get (WYSIWYG) Web 2.0 editor, any user can change the content of the hybrid wikis without any HTML or XML knowledge. Further, WYSIWYG editor allows users to add visualizations (Figure 5.20). The user can form the visualization by clicking on the visualization creation icon in WYSIWYG editor. This action opens a web based visualization configurator that guides users to create generic visualizations.

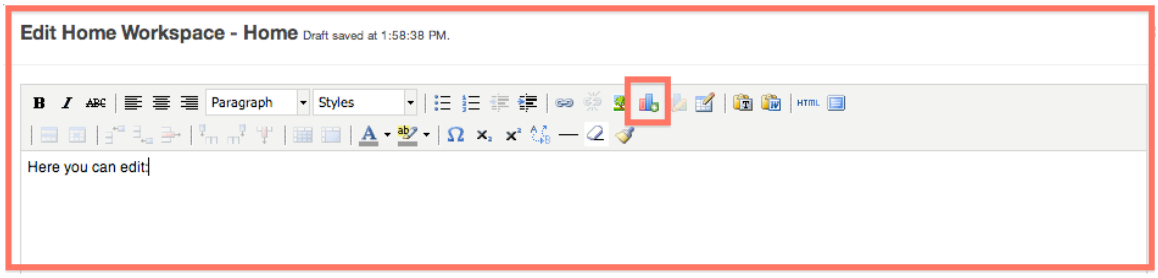


Figure 5.20.: WYSIWYG Web 2.0 Editor.

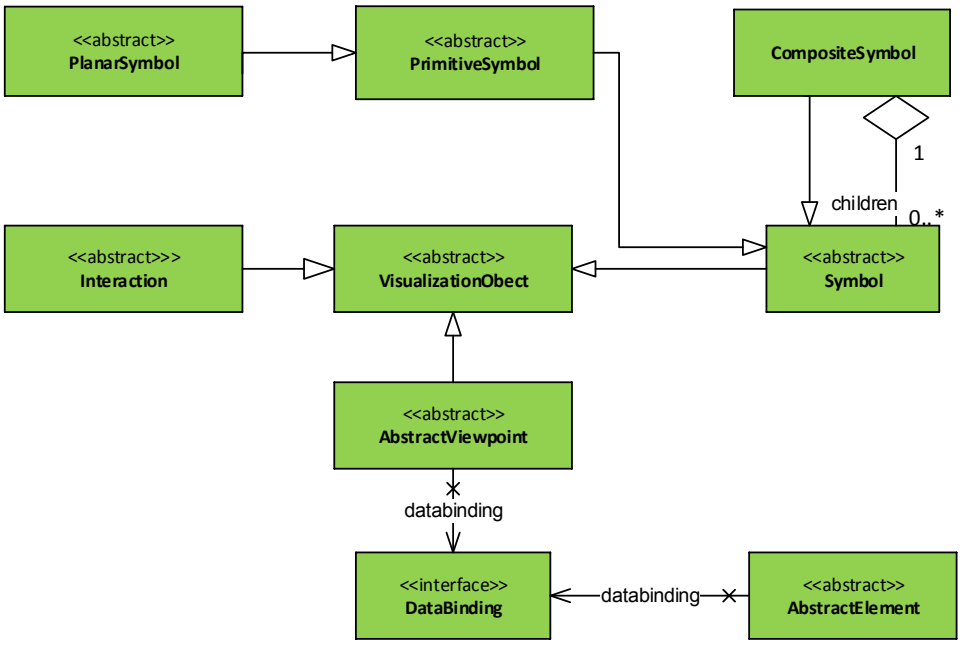


Figure 5.21.: Generic class diagram.

Figure 5.21 represents a generic class diagram to explain the basic structure of visualization creation in Tricia. The following class diagram explains how to create viewpoints in Tricia. In the class diagram, **Abstract Element** class includes abstract visualization element. It can be named as a visualization pattern that is independent of the data model of the visualization. The end user can choose the viewpoint that is present in Tricia from the first page of the visualization configurator (Figure 5.22). Every Abstract Element should have a data binding to bind the information model to the abstract visualization (Figure 5.23). The class called **DataBinding**, which inherits two classes called **HybridDataBinding** and **DummyDataBinding** have the data binding. DummyDataBinding binds hard coded information model to the abstract visualization. However, in the HybridDataBinding the pattern-matching algorithm works to match information model to the abstract visualization. In order to have successful hybrid data binding configuration, users should filter

the data model to narrow down the information model. This filter mechanism produces a JSON string for each selected filtered information model (Figure 5.24). This JSON string is parsed during the hybrid data binding, and it gives the result according to selected filter string. According to the filter string, filtered information model is mapped to the abstract visualization element. If the sub information model that is produced by the filter mechanism does not correct then the user can not select the proper mappings. In other words, this filter mechanism forces user to select correct sub information models.

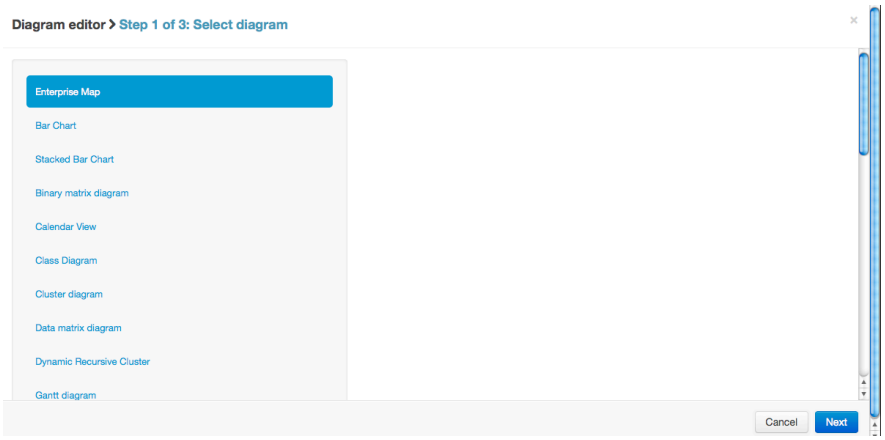


Figure 5.22.: View point selection

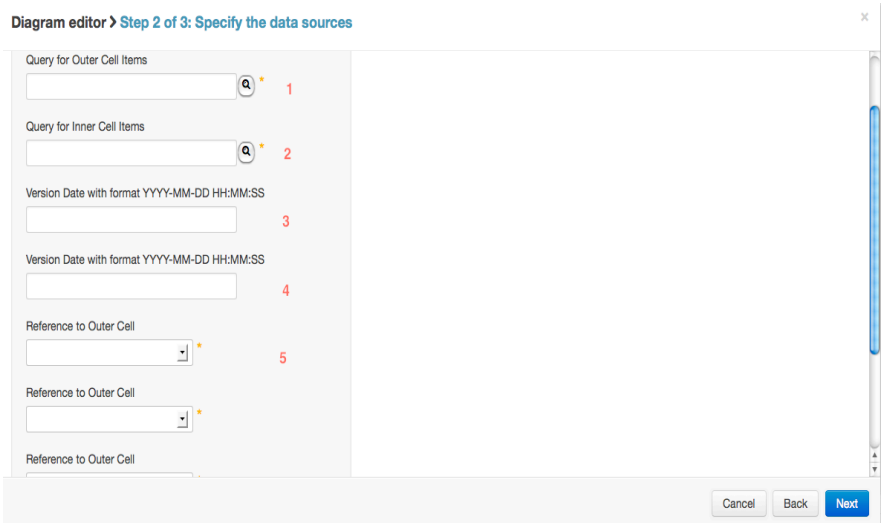


Figure 5.23.: Databinding to viewpoint

{"all":[{"spaceId":"15ky4xxlrfgkv"},
{"contentType":"page"},
{"type":"country"},
{"attribute":"name"}]}

Figure 5.24.: Sample Data Filter String

AbstractViewPoint class is an abstract viewpoint, which realizes the data binding. It also handles the shape drawing according to the data binding of the abstract visualization. This class first handles the databinding and afterwards allows user to configure the visual elements (Figure 5.25).

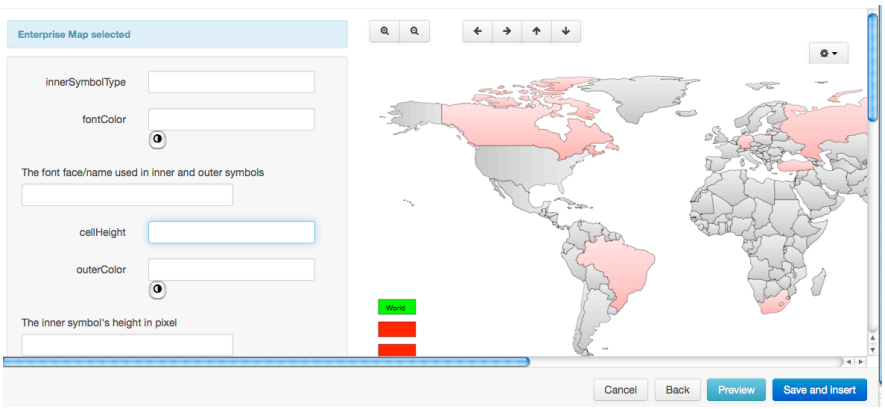


Figure 5.25.: AbstractViewPoint configurator

Figure 5.26 gives an overview about the information model that will be used to create the following visualizations. The whole data model was presented before, however for the implementation purposes, only some parts of it has used because visualization logic behind other elements are very similar to the filtered one. This data model will be used heavily in the data binding of the visualizations.

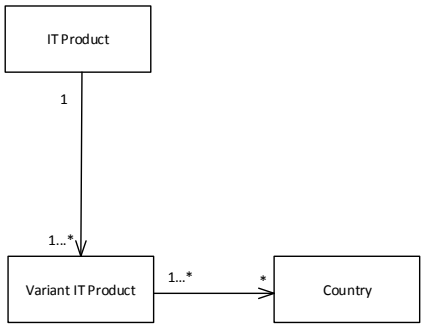


Figure 5.26.: Information model

The following subchapters will discuss the implementation of different visualizations by using the given information model.

Figure 5.27 is a basic class diagram that shows the symbol and the interaction, which are two important visualization primitives in Tricia. This class diagram is not representing all list of the interactions and symbol, but only the relevant ones for this research. Symbol is the abstract class for the visualization primitives like line, rectangle or any other ones. Interaction is the abstract class that handles the entire user event like mouse click event, hover etc. CompsiteSymbol and CompositeInteraction enable users to have a list of interactions or symbols. This allows Tricia to define a compact visualization that has many visualization primitives and set of sequential events.

In order to render the visualizations and interactions in the web browser, Tricia is using Raphael framework, which is a JavaScript library that enables programmers to create easily, customized vector graphics and interaction on the web browser [6]. Figure 5.28 is a general overview of this rendering process. Any viewpoint in Tricia binds visualization model with the data schema. Visualization model includes the visualization primitives that are introduced in Figure 5.27. In order to render the viewpoint visualization, model calls the renderer package which sets the attributes of the defined interactions and visualizations and and transfer to Raphael.stg file. This file is responsible with the correct mapping between the Java code and the Raphael framework. Raphael framework imports the file and generates the JavaScript code for the browser. Figure 5.29 is a sample of the basic code sample to define the interactions and symbols inside the Tricia.

[6] `https://http://raphaeljs.com/`; visited on September 20th 2013.

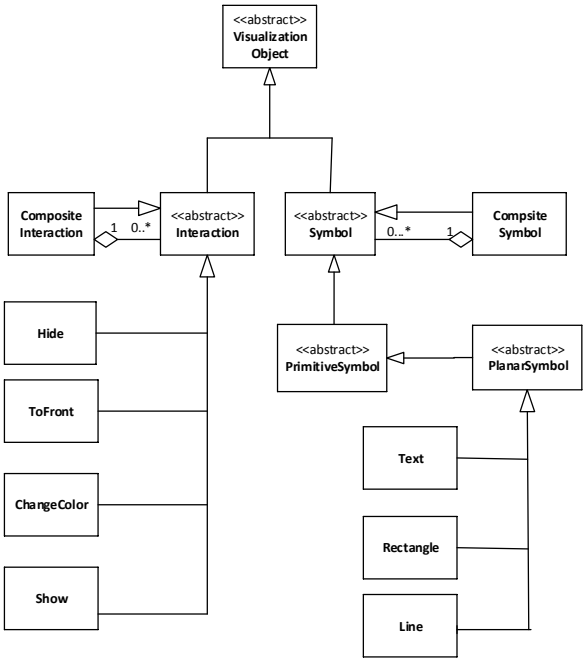


Figure 5.27.: Class diagram for Symbol and Interaction

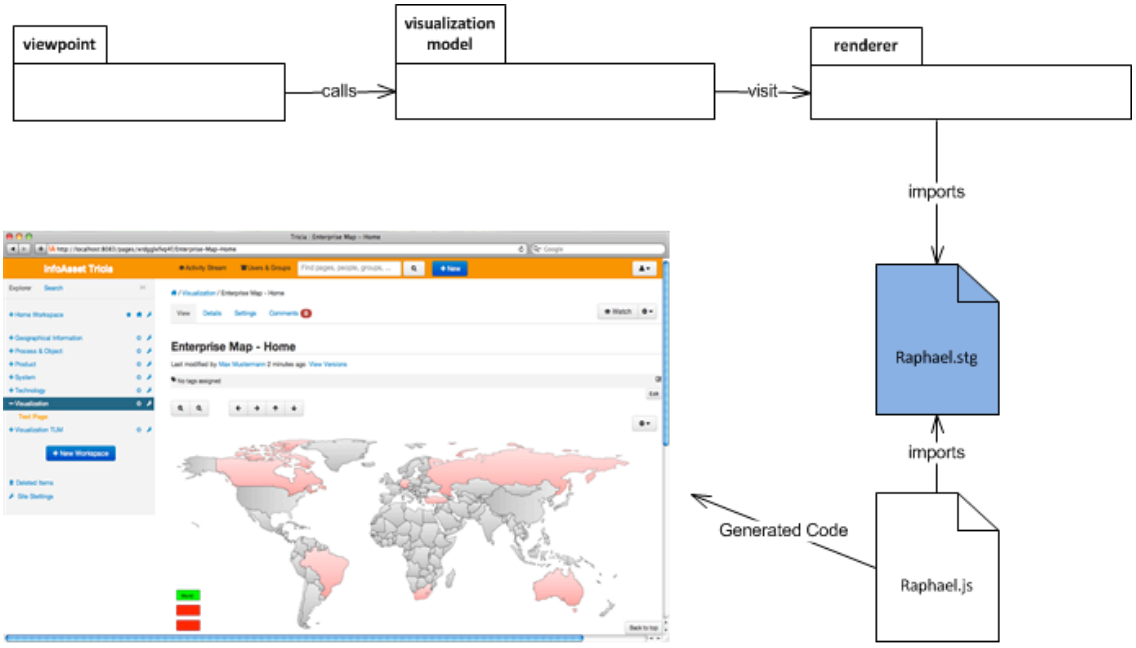


Figure 5.28.: Package diagram of visualization generation with Raphael

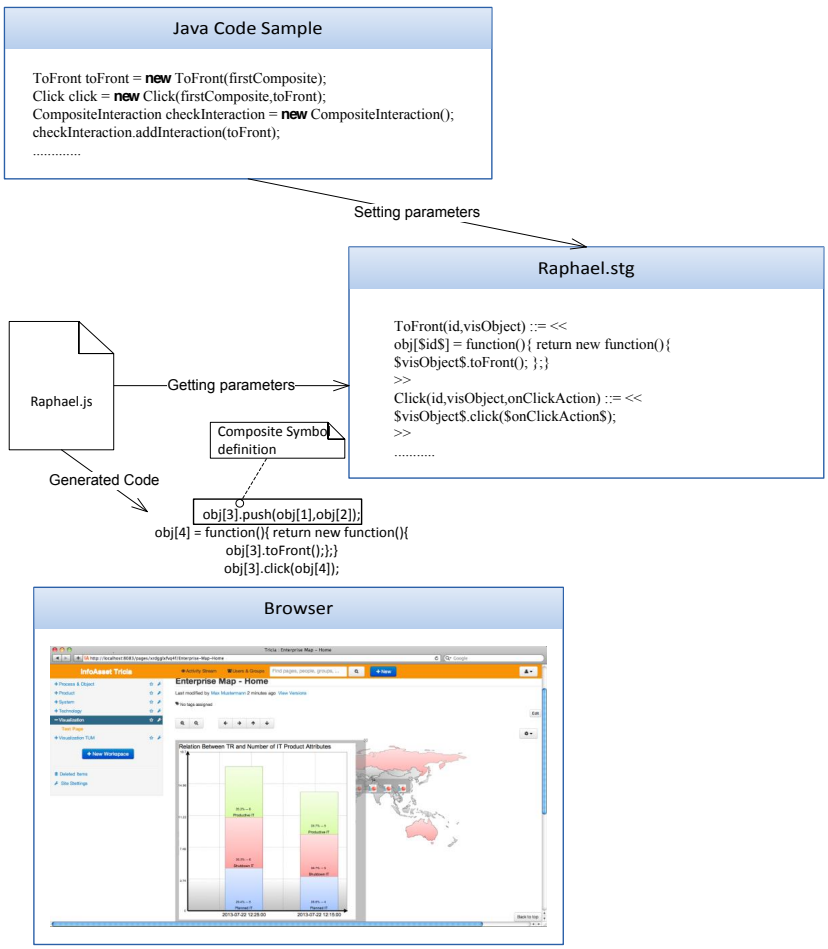


Figure 5.29.: Code sample of visualization generation with Raphael

5.5.2. Stacked Bar Chart

As a first step stacked bar chart visualization was implemented. To get a better understanding of the visualization, Figure 5.30 gives an overview between the mapping of abstract visualization concept and information model. Stacked bar chart consists of many inner bar charts according to the number of references. These references explain the relation between the outer bar chart and inner bar charts. It is used for grouping the information based on their data similarities. For example, these references can be state of the inner bar chart like planned, shutdown or productive. Figure 5.30 explains three different state of IT Variant with the Country based on different version information. The first stacked bar chart shows the first version of the given country with three different relation between IT Product Variant.

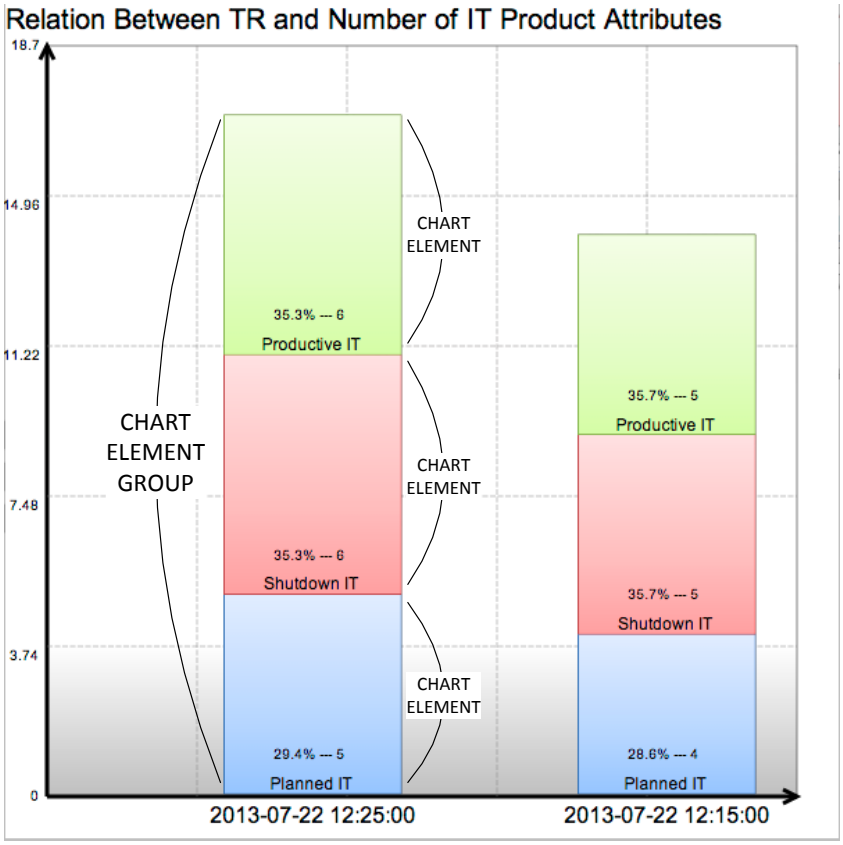


Figure 5.30.: ChartElementGroup , ChartElement

Figure 5.31 [7] is a basic UML class diagram of the stacked bar chart visualization. As stacked bar chart consisting of inner bar chart, there is **ChartElementGroup** class, which contains all different inner charts. **ChartElement**, which inherits AbstractElement class, is a basic abstract element that has the abstract visualization called inner charts.

[7] Viewpoints called ClusterMap explained in detail in Figure 5.34

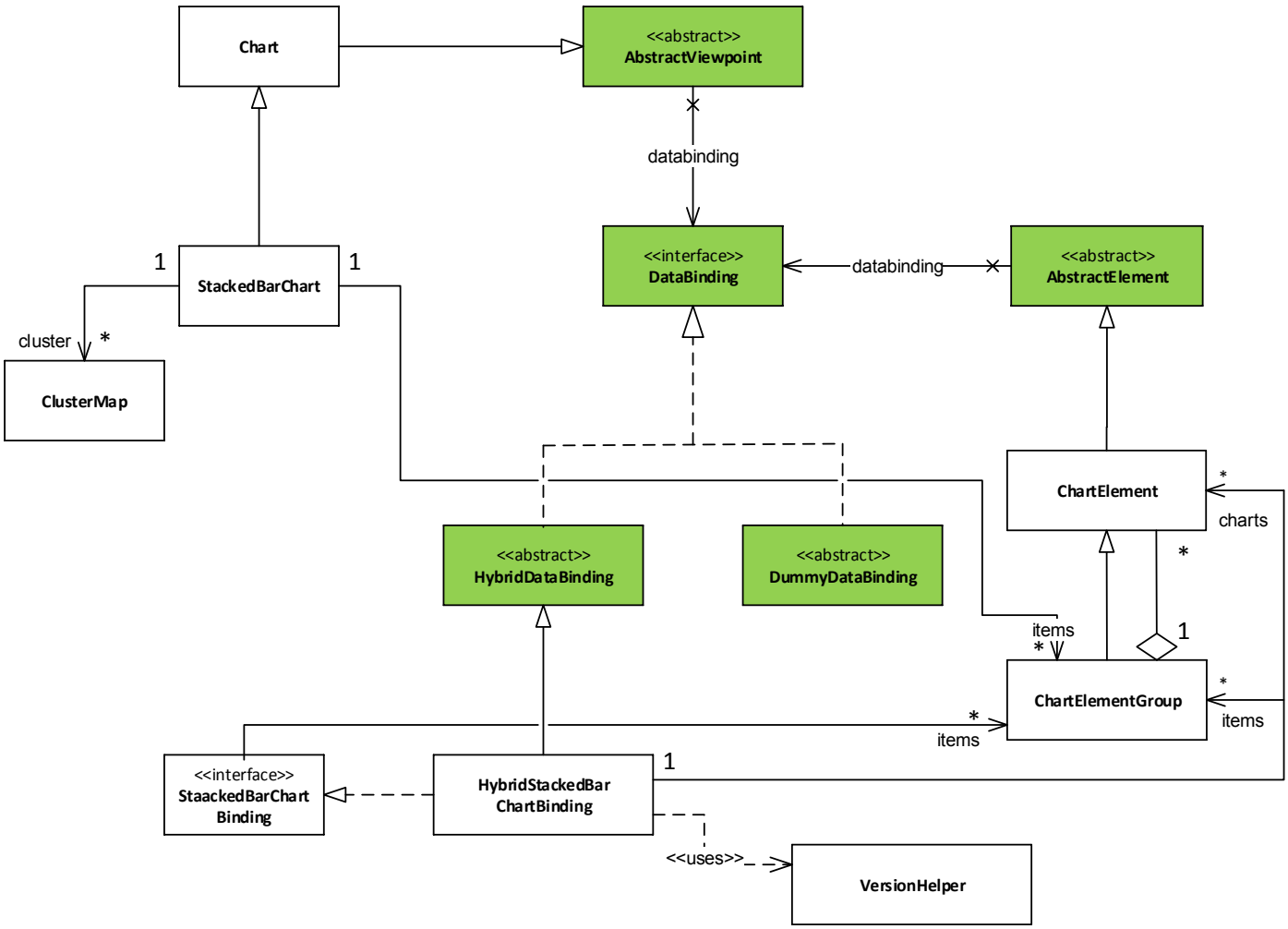


Figure 5.31.: Class diagram for stacked bar chart

HybridStackedBarChartBinding is a data binding class, which binds abstract visualization model to the information model. The end user can determine this configuration by having the correct mappings in the visualization configurator (Figure 5.32). The end user can select outer bar chart (ChartElementGroup) (Figure 5.32 (1)) and inner bar charts (ChartElement) (Figure 5.32 (2)) by using the filter mechanism of Tricia. Then the pattern-matching algorithm works and offers possible attributes to visualize between the outer and inner bar charts (Figure 5.32 (4)). After entering the desired version dates, the data binding class calls VersionHelper class to get different versions from the ChangeSet schema (Figure 5.32 (3)).

Figure 5.32.: Visualization Configurator for stacked bar chart

StackedBarChart class is the AbstractViewPoint class, which calls the data binding, visual elements and interactions. After having the data binding inside StackedBarChart, abstract elements are transformed to the real visualization by the color and shape preferences of the end user in visualization configurator.

5.5.3. Cluster Map with Colour Coding

Cluster map is another visualization that is developed both conceptually and implementally. The idea behind the cluster map with color-coding is explained in depth in the conceptual part. Before, cluster map visualization has already implemented in Tricia. In the context of the master thesis, Cluster map is extended in a way that it can visualize different attributes in a single cluster map. In order to accomplish this, color-coding is used. Likewise in the stacked bar chart visualization, each color represents different attributes of the visualized elements. Mapping between the abstract visualization model and information model is similar to the stacked bar chart (Figure 5.33). According to the figure, there is one outer element (Figure 5.33 (1)) contains many inner elements which represent the different attributes of them by using color coding (Figure 5.33 (2)). The following figure shows IT variants of particular country based on different relations.

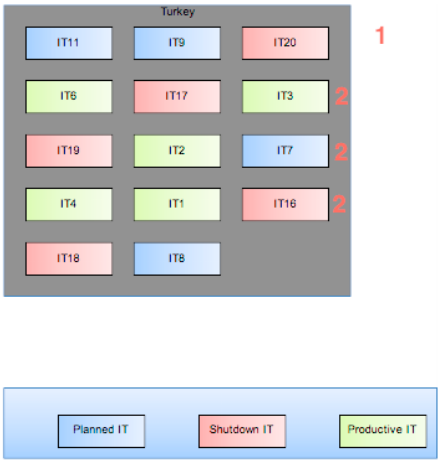


Figure 5.33.: Cluster Map with Color Coding

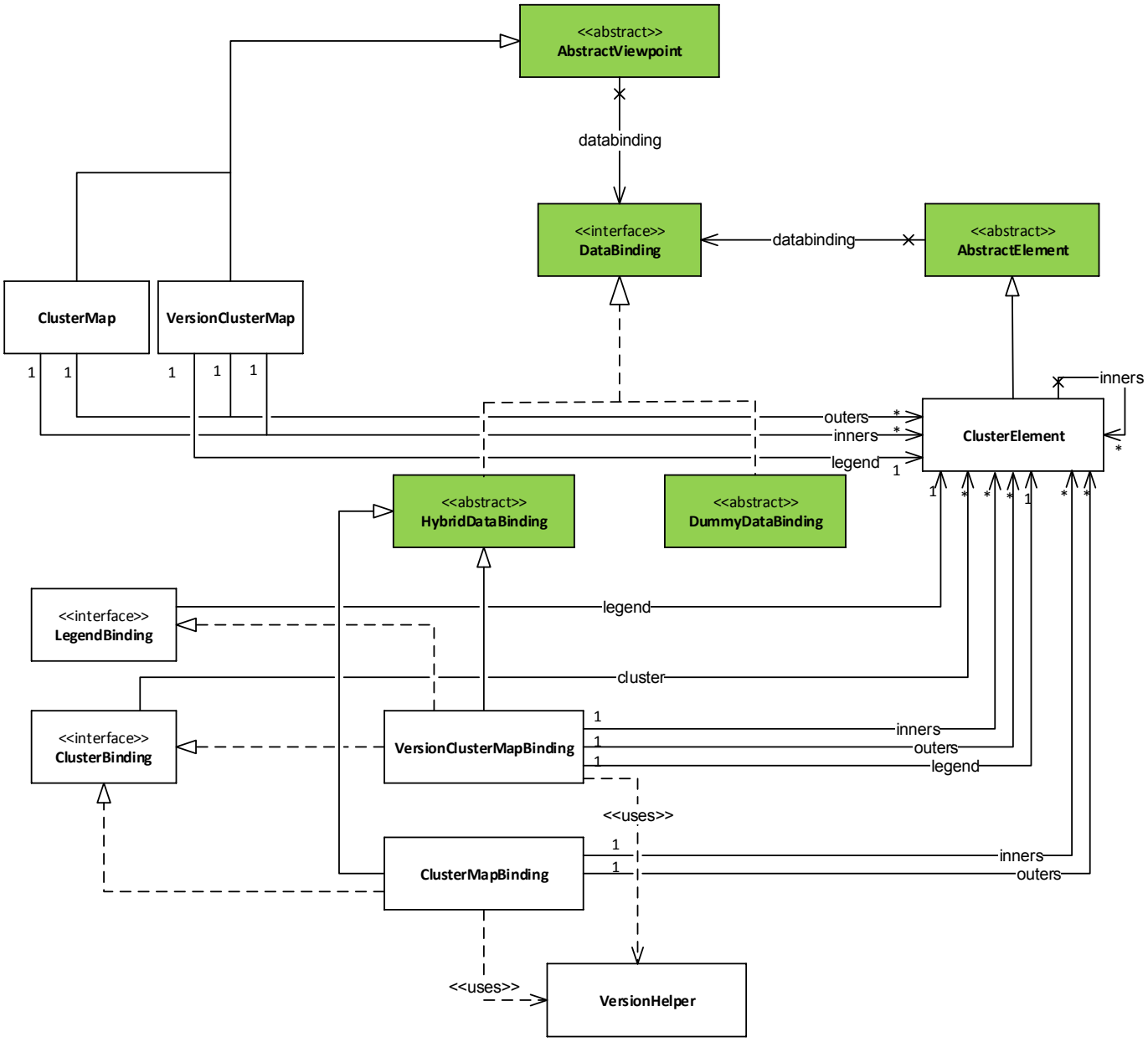


Figure 5.34.: Cluster Map class diagram

5.5.4. Layered Cluster Map with Colour Coding

Layered Cluster Map is an alternative representation of the set of Cluster Maps. For detailed information behind the layered cluster map, please check the conceptualization part.

Differently from the Cluster Map implementation, Layered Cluster Map has ClusterLayerGroup class which has many different ClusterElement objects. ClusterLayerGroup class

is used to represent an each layer in the Layered Cluster Map (Figure 5.35). In order to illustrate a specific layer upward, this layer is clicked. As a result, previously showed and desired layer are swapped. Just like as in the symbol, the Interaction also handles by the Raphael library.

Layered cluster map aims to show different cluster maps in a more compact representation. It is a space efficient visualization that build layers for different cluster maps. It is very convenient to use especially when the data gets too big. One sample scenario example about this visualization will be given in the interactive map visualization subsection.

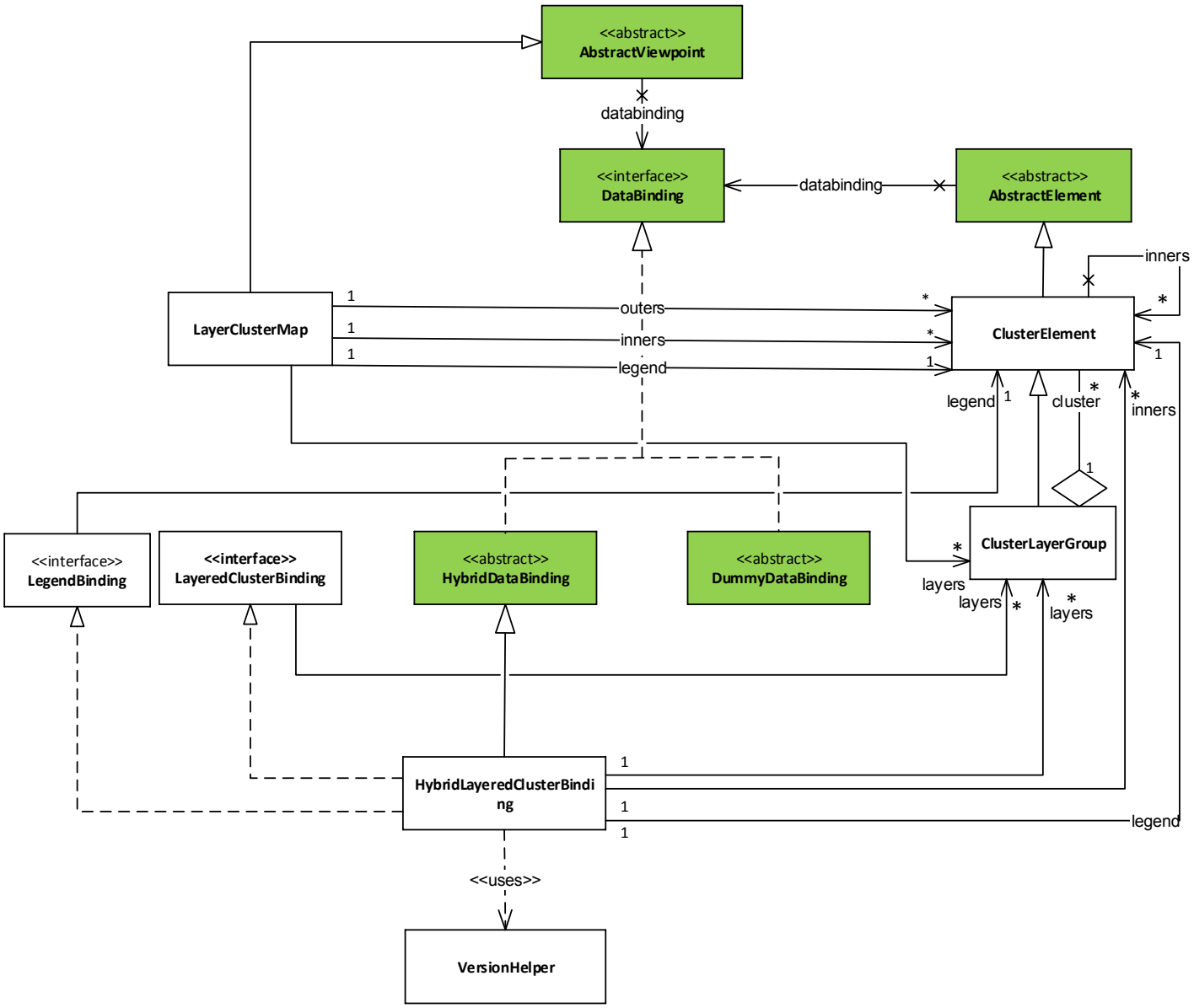


Figure 5.35.: Layered Cluster Map class diagram

5.5.5. Interactive Map Visualization

Interactive map visualization is a combination of map and some implemented visualizations that were introduced before. It is based on the world map in which it aggregates visualizations like bar chart and cluster map into it.

Figure 5.36 [8] [9], illustrates the class diagram of Interactive map visualization. As it is observed from the class diagram, HybridEnterpriseMapBinding is a class, which inherits its HybridDataBinding in order to provide data binding. However, this data binding is different from other visualization since it does not generate any data binding with AbstractElement class. Instead of it, it only has the data binding configuration parameters and passes them to the AbstractViewPoint class. HybridEnterpriseMap imports all other AbstractViewPoint and passes the determined data-binding configuration (Figure 5.36).

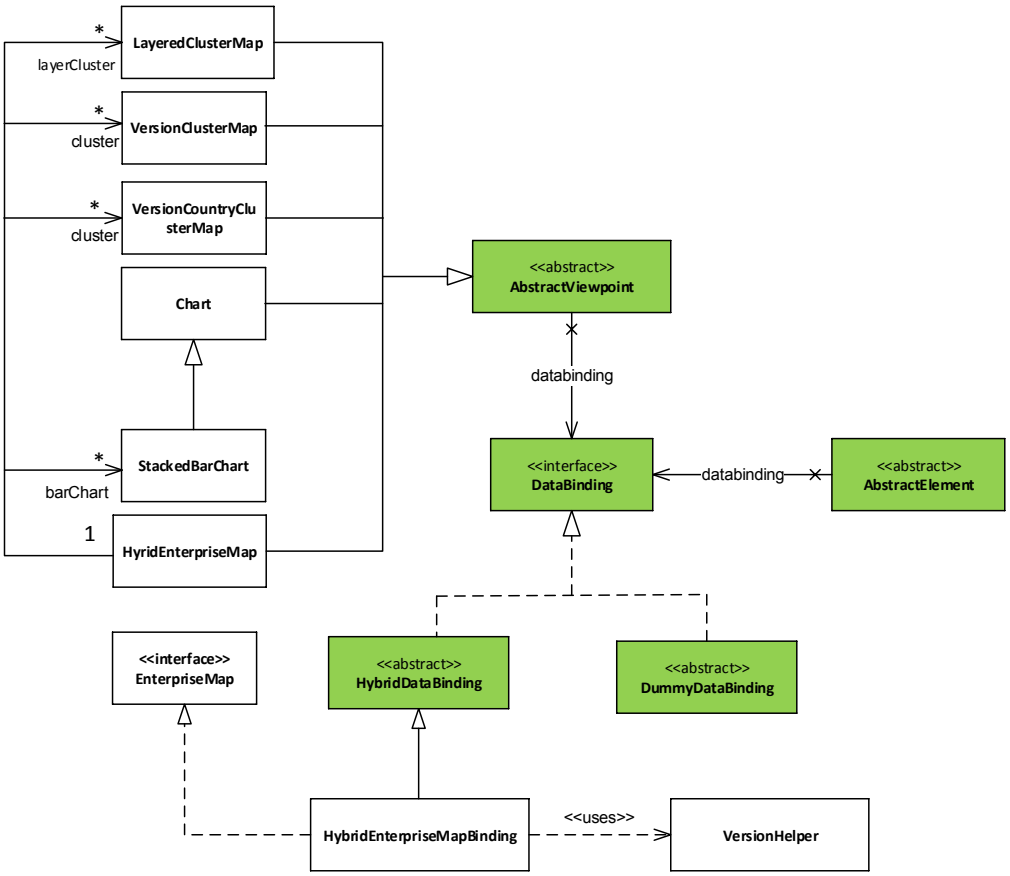


Figure 5.36.: Interactive Map visualization class diagram

[8] Viewpoints called LayeredClusterMap,VersionClusterMap,VersionCountryClusterMap explained in detail in Figure 5.35 and Figure 5.34

[9] Viewpoints called StackedBarChart explained in detail in Figure 5.31

Figure 5.37 represents an overview of Interactive Map visualization. According to the information model, this visualization shows the relation between IT Product Variant and Country. For this visualization, there are different view options. For instance, the user can either select to view the Map visualization with world map option (Figure 5.38 (1)) or cluster map options(Figure 5.38 (2), (3)). The world map uses color-coding in order to indicate the active countries, that has IT Product Variants inside (Figure 5.38 (4)). When user clicks on one of these active countries, an interactive menu displays the country specific visualizations for the selected country (Figure 5.38 (6)). The user can easily visualize the different visualizations of the selected country by clicking each item in the menu (Figure 5.40, 5.41, 5.42). To make the world map fancier, another color is used to show the active country by using mouse over action (Figure 5.38 (4)).

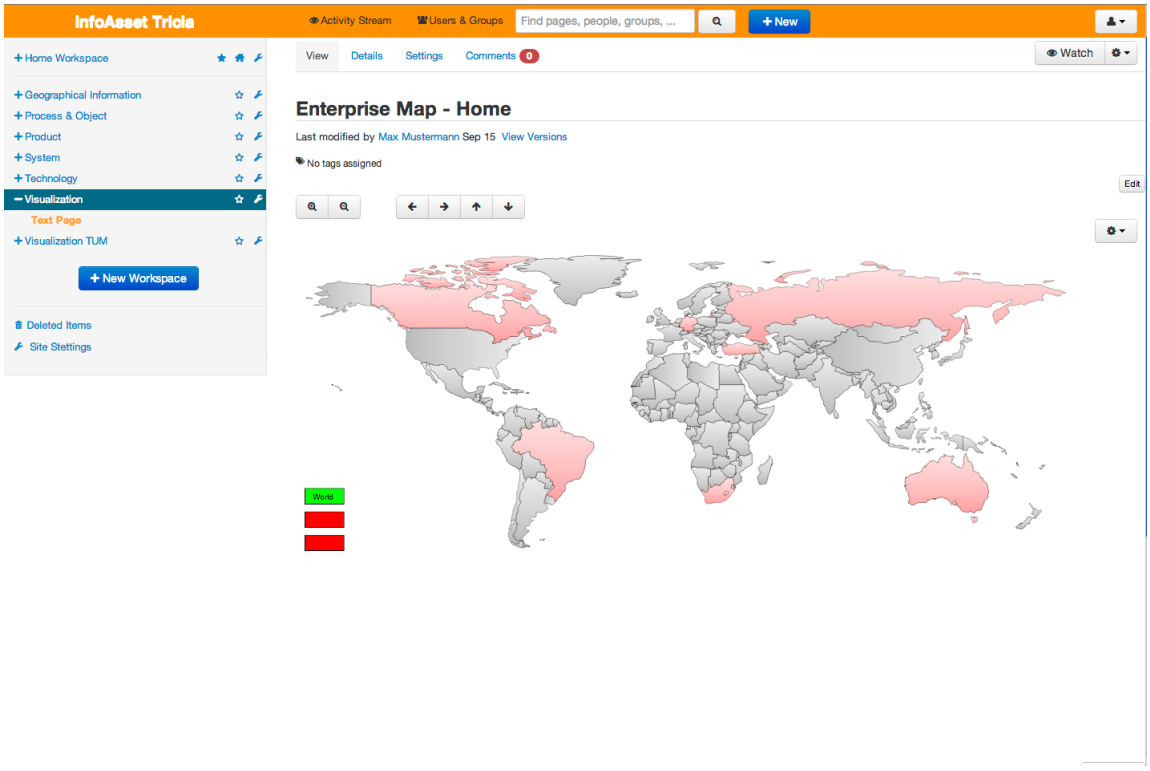


Figure 5.37.: Interactive Map visualization

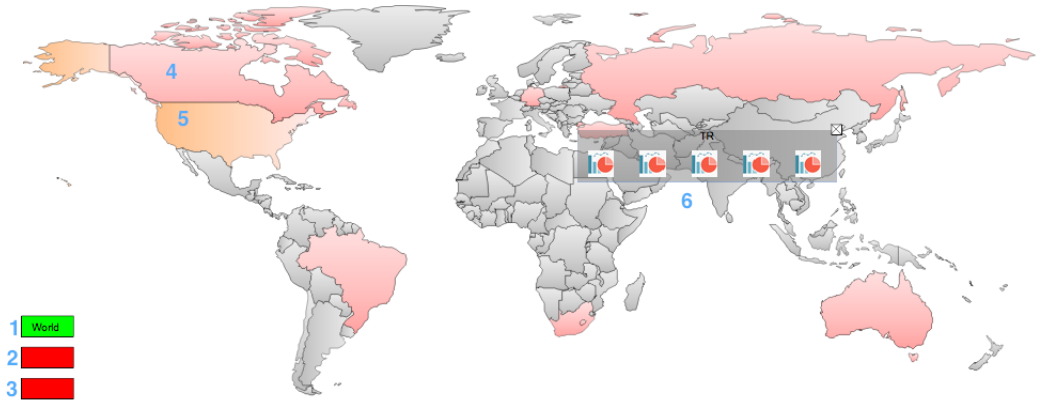


Figure 5.38.: Interactive Map visualization explanation

The interactive map visualization offers another options that user can select to preview different versions of whole portfolio by cluster map (Figure 5.38 (2), (3)) (Figure 5.39). It contains the same information as the world map but not in map form. By clicking on the buttons that is shown on the left side of the map, the user can visualize status of different active countries. Figure 5.39 shows one example that visualizes the one version of active countries. The advantage of cluster map view option against the map view is, the user can view the whole portfolio at one glance.

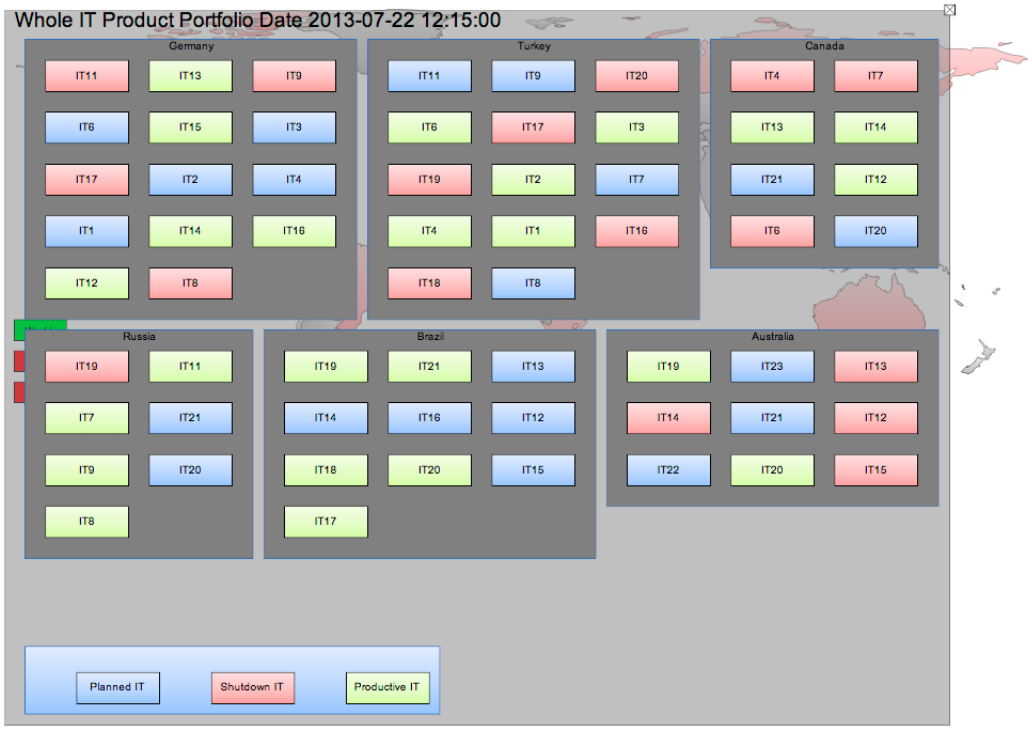


Figure 5.39.: Interactive Map visualization with cluster map whole portfolio

As it is stated before, the active countries on the world map are distinguished with color-coding. These active countries have inner visualizations which enables end users to observe these countries in more detail. Each active country has a click action which opens an interactive menu that shows the country specific visualizations. For example, the first item in the visualization menu displays the stacked bar chart which shows the number of IT Product Variant for the active country (Figure 5.40). Each stacked bar chart shows different versions by having different IT Product Variant states as inner charts.End users may get curious about the content of the each inner chart inside the stacked bar chart. In order to accomplish this end user should click on the desired inner chart. The detailed content of that inner chart is represented by having the extra cluster map.

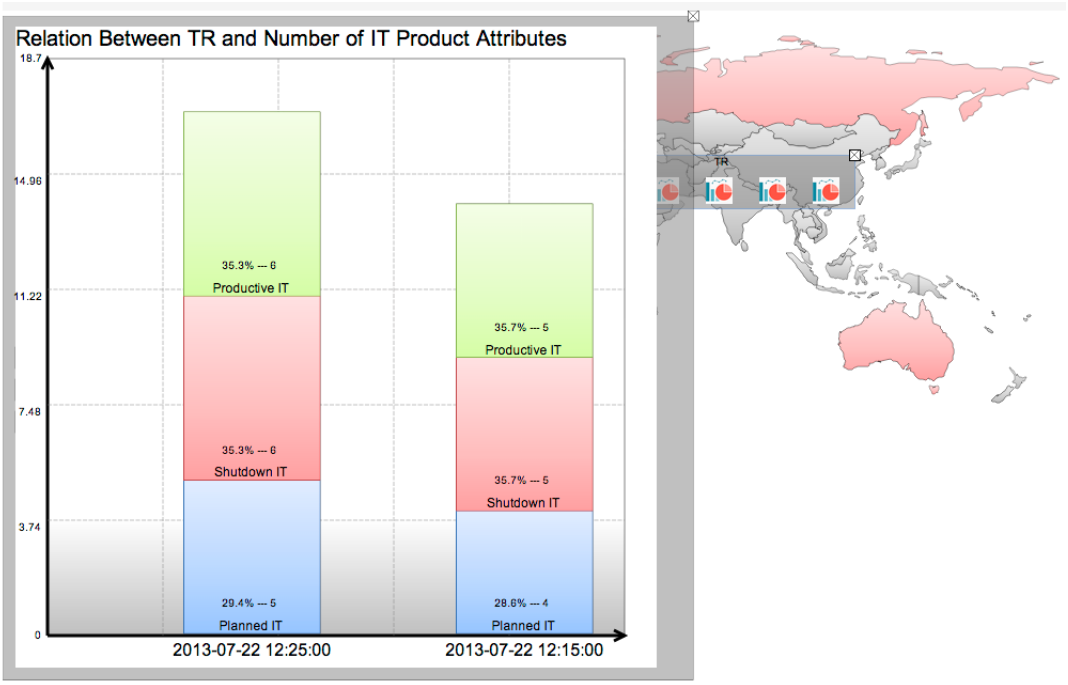


Figure 5.40.: Interactive Map visualization with country specific stacked bar chart

Figure 5.41 shows layered cluster map, which is activated by clicking the second item in visualization menu. This visualization aims to show the relation between IT product variant and country. Each relation between these two objects is identified by each layer. Each layer shows the set of IT product variants with the selected relation by including the version information. For instance, figure 5.41 shows as a first layer set of Planned IT in the selected country. Transitions between the layers are accomplished by the click action. According to this action, the clicked layer come to the front, and the one on the front goes to the tail. Moreover, when the user does not want to use the animation of the layered cluster map, the interactive map visualization also offers cluster maps per each relation. Figure 5.42, figure 5.43, figure 5.44 represents the sample visualizations which shows the cluster map without layer option. Layers in the layered cluster map visualization are distributed to three different figures. For instance in the given figures, each button offers one layer of the pre visualized layered cluster map.

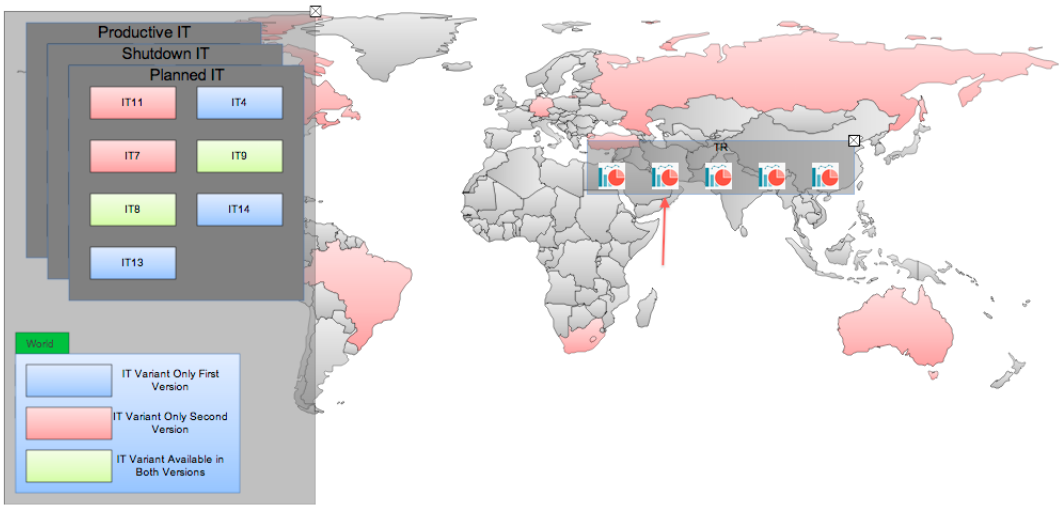


Figure 5.41.: Interactive Map visualization with country specific layer cluster map

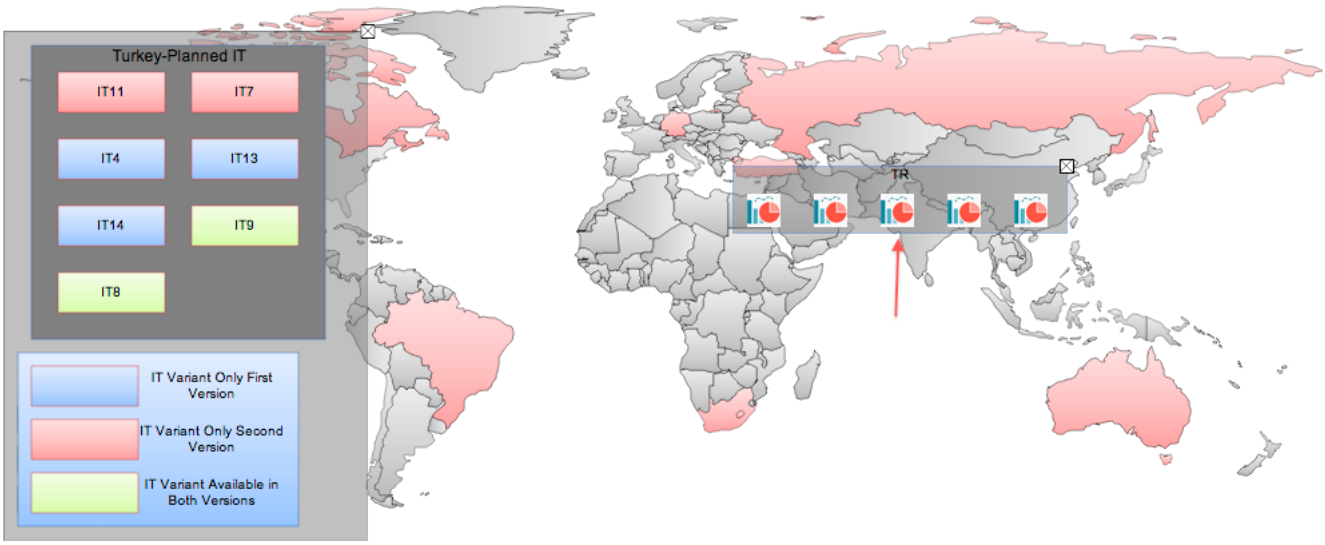


Figure 5.42.: Interactive Map visualization with country specific cluster map

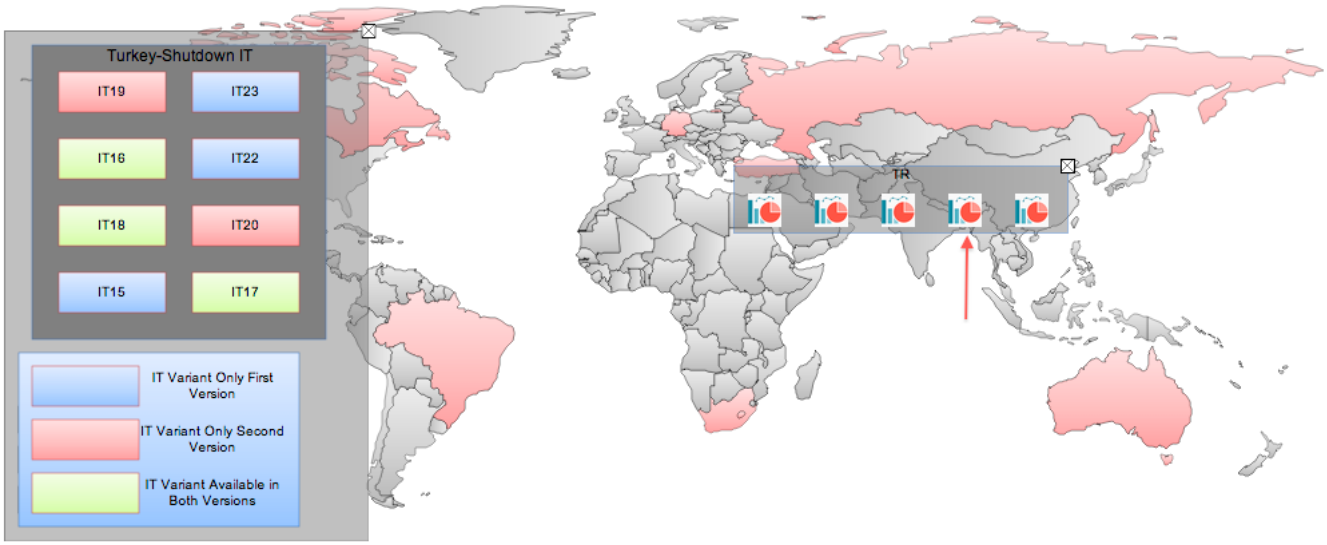


Figure 5.43.: Interactive Map visualization with country specific cluster map

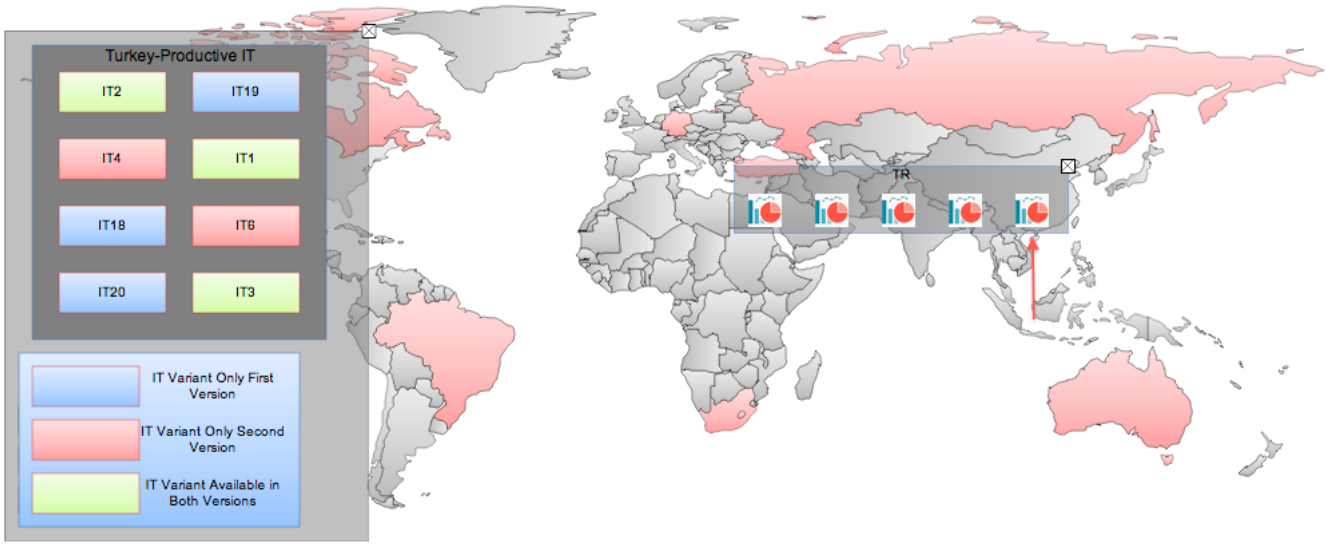


Figure 5.44.: Interactive Map visualization with country specific cluster map

Part IV.

Evaluation and Future Work

6. Evaluation

6.1. Final Interview

The following sub chapter consists of final questionnaires to verify the concept of this research. These final questionnaires are a structured interview with the stakeholder consisting of several sub-parts. Besides evaluating the conceptual work of this research, the interview contains some questions to have beneficial feedback about the platform called Tricia.

6.1.1. Import Mechanism Questions

1. How easy is to prepare the import mechanism file?

   a) Very Easy - Easy - Moderate - Hard - **Very Hard**

   b) What are the challenges for the import mechanism file?

   Answer
   You need to have a deep knowledge about Tricia and corresponding XML structures of the import mechanism. It needs an intensive training.

2. How easy is to import data?
   Answer
   It is not quite easy to use. The reason is, we do not have a stable version of Tricia.

6.1.2. Visualization Questions

Stacked Bar Chart

1. Do you think it fulfils the given requirement(Table (3.10))?
   Answer
   Yes, it fits to the requirement.

2. Can you easily recognize the differences between the versions?
   Answer
   Yes, different attributes and differences between the versions are very easily recognizable.

3. Does the information enough to explain the visualization?
   Answer
   Yes,it is very informative and self-explanatory with the writings inside inner charts.

4. When you compared it with your current solutions, does it improve your solution?
   Answer
   Yes, it improves our current approaches to the problem.

Cluster Map

1. Do you think it fulfils the given requirement(Table (3.11))?
   Answer
   Yes, it fits to the given requirement.

2. How easily can you detect the different attributes?
   Answer
   It is very easy to compare and detect different attributes.

3. Does the information enough to explain the visualization?
   Answer
   Yes, especially with the legend, it is very intuitive to use and explore.

4. When you compared it with your current solutions, does it improve your solution?
   Answer
   Yes, it improves our current approaches to the problem.

Layered Cluster Map

1. When you compared it with the normal cluster map, do you think this solution is better?
   Answer
   It is more confusing. Extra layers do not contribute for the extra functionalities. Cluster Map without any layer is more intuitive to use.

2. Can you easily distinguish different attributes and states?
   Answer
   Many layers distract attention. Differences between the states and attributes are confusing.

3. Does the information enough to explain the visualization?
   Answer
   Yes, it is enough.

4. When you compared it with your current solutions, does it improve your solution?
   Answer
   Yes, but plain cluster map is better to use.

Cluster Map with Time Line

The interviewer gave general responses about this visualization. According to him, this visualization is similar to the Layered Cluster map and it is confusing. The interviewer preferred to have normal cluster map.

Map Visualization

1. Can you easily realize the countries which have the visualizations?
   Answer
   Yes, the countries and their visualizations are obvious in the visualization.

2. Are the visualizations that are specific to countries understandable?
   Answer
   Yes, but the menu for the countries visualizations has room for improvement. The same menu item for different visualizations is confusing. Furthermore, different visualizations for the specific country are not easily recognizable.

3. Do you think in overall, is the concept in map visualization logical?
   Answer
   Yes, it is very logical with a good concept.

4. Does it give you a sense of dashboard or a management board?
   Answer
   It gives an overall concept of the given requirement. It can be used as a dashboard.

5. When you compared it with your current solutions, does it improve your solution?
   Answer
   Yes, it can improve our current system.

Tree Structure Visualization

1. Do you think it fulfils the given requirement(Table (3.4))?
   Answer
   Yes, it fits with the given requirement.

2. Can you easily realize standard technologies?
   Answer
   Yes, but as the size grows, it is harder to detect standard technologies. Furthermore, there is a scalability problem.

3. Can you realize the number of different technologies that are standard per category? Can you please give some examples that you have detected in the first sight?
   Answer
   Yes, but again there is a scalability problem. The examples are detected, however it is confidential.

4. Can you detect the differences between the different versions?
   Answer
   Comparisons of the different versions are very hard. To compare different versions, the whole shape should be compared with other, which is very hard and error prone.

5. When you compare it with your current visualization, does it improve your approach or solution?
   Answer
   No, it does not improve our current approach and solution.

6. What are the limitations?
   Answer
   Scalability is one of the biggest limitations of this visualization. When the number of items increases, the visualization gets bigger and bigger which causes problems.

7. What can be the further improvements?
   Answer
   Collapse and expand mechanism can be added per item to decrease the complexity and size of the visualization.

8. Do you think that would it be worth to develop this kind of visualization?
   Answer
   No, it does not improve our current approach and solution.

FanLens

1. Do you think it fulfils the given requirement(Table (3.4))?
   Answer
   Yes, it fits with the given requirement.

2. When you compare it with the Tree Structure Visualization, does it improve the solution?
   Answer
   Both of the visualizations have some advantages, when expand and collapse mechanisms are added to the tree structure visualization. FanLens can be used to have structure-based collapse and expand mechanisms. On the other hand, tree structure may still visualize the whole overview with the collapse and expand mechanism.

3. When you compare it with your current visualization, does it improve your approach or solution?
   Answer
   Yes, it improves our current solutions.

4. Do you think that would it be worth to develop this kind of visualization?
   Answer
   No, this visualization will not bring significant improvement to our current solution.

Spie Chart

For this visualization, the stakeholder stated general objectives. For the stakeholder, Spie chart is a very confusing and unintelligible. The visualization needs too much time to understand. Stakeholder is not favour of using this visualization because of its limitation and complexity.

Updated Circle View

In overall, stakeholder assessed this visualization as a suitable one. According to the stakeholder, this visualization fits the requirement and it is nice to compare different versions and attributes. However, the stakeholder prefers to use stacked bar chart for comparing different states and attributes.

Radar Plot Visualization

1. Do you think it fulfils the given requirement(Table (3.5))?
   Answer
   Yes, it fits with the given requirement.

2. Can you realize the differences between different states?
   Answer
   Yes, different states are easily recognizable.

3. When you compared it with your current solutions, does it improve your system?
   Answer
   Yes, it improves the current solution of the enterprise.

4. Do you think that would it be worth to develop this kind of visualization?
   Answer
   It is a very well known visualization among the business people and it is very easy to understand. Furthermore, it is worth to develop this visualization.

5. When you compared Spie Chart, Updated Circle View and Radar Plot which one do you think is the most suitable one?
   Answer
   Because of the level of recognition, Radar Plot is preferred among these three visualizations.

Circos for Business Visualization

1. When you compared it with your current solutions, does it improve your system?
   Answer
   No, it does not improve the current solution of the enterprise because of its limitations. Readable results are not expected from this visualization.

2. What are the limitations?
   Answer
   In general, it is a fine visualization; however, it is very restricted when it comes to business visualization. Readability of the visualization is low, when the number of data increases.

3. Do you think that would it be worth to develop this kind of visualization?
   Answer
   The visualization does not produce suitable business visualizations with a huge amount of data. Furthermore, it does not worth to develop this visualization.

Matrix Representation

1. Do you think it fulfils the given requirement(Table (3.6))?
   Answer
   Yes, it fits to the given requirement.

2. Can you realize the mapping between the product and properties?
   Answer
   Yes, the mapping between the product and properties can be realized easily.

3. Can you realize the differences between different versions?
   Answer
   Yes, differences between different versions are very obvious and easy to detect.

4. When you compared it with your current solutions, does it improve your system?
   Answer
   No, it does not improve solutions in the enterprise. The enterprise already has this kind of mature visualizations. However, when it comes to historical visualization, it improves the current solution since the enterprise does not have this kind of visualization.

5. What are the limitations?
   Answer
   It does not seem to have any limitation of the visualization.

6. Do you think that would it be worth to develop this kind of visualization?
   Answer
   Yes, but it should be developed with a historical visualization.

Version Mechanism

1. Have you ever tried to create different versions?
   Answer
   Yes, together with the guidance.

2. How easy is to create different versions?
   Answer
   Very Easy - Easy - Moderate - **Hard** - Very Hard

3. Are there any limitations in the version mechanism?
   Answer
   Right now, it is in experimental state. Having an accomplished version mechanism is hard right now.

4. Does it need any other further development?
   Answer
   Yes, Tricia should get more enterprise ready product. The current version that we were using in this thesis work, was not a stable and reliable version. The guidance and documentation are very limited.

Visualization Creation

1. Have you ever used the visualization configurator?
   Answer
   Yes, together with the guidance.

2. How user friendly is the visualization configurator?
   Answer
   The configurator is very hard to use especially for the end users. The explanation of the internals of Tricia is missing. Thus, usage of the visualization configurator becomes harder.

3. What is the limitation of visualization configurator?
   Answer
   The biggest limitation of visualization configurator is the lack of documentation.

4. How hard is to create visualizations?
   Answer
   Very Easy - Easy - Moderate - **Hard** - Very Hard
   It is hard because of the lack of documentation.

Which one of the visualizations you may use in the future?
Answer

Stacked Bar Chart	✓
Cluster Map	✓
Layered Cluster Map	✗
Cluster Map with Time Line	✗
Map Visualization	✓
Tree Structure Visualization	✗
Fan Lens	✗
Spie Chart	✗
Updated Circle View	✗
Radar Plot Visualization	✓
Circos	✗
Matrix Representation	✓

6.1.3. Overall Questions

1. Have you ever used the Tricia tool?
   Answer
   Yes, with the guidance.

2. When you compared your system with Tricia, what do you think about the strengths of your system?
   Answer
   Stability, performance issues, easiness of manipulating and creating the visualizations are mainly strengths.

3. When you compared your system with Tricia, what do you think about the limitations of your system?
   Answer
   Tricia has a version comparison system. Tricia is able to create historical visualization

and can compare different EA model states. Moreover, Tricia has hybrid wiki pages system, which enables user to explore the system without much effort.

4. What can be the further development and future research development areas?
   Answer
   Version mechanism and historical visualizations can be a suitable candidate for the further development. Active filtering mechanism of visualizations by the end users can be another topic. Also, optimization of Tricia can be determined as a future research development area.

In overall, the stakeholder found the concept useful and beneficial. The following information includes the summary of the feedbacks from the stakeholder.

For the import mechanism, the stakeholder found it very hard to realize because of the documentation issues and maturity of the product. This problem will be solved mostly after introducing this thesis since the import mechanism is introduced comprehensively. According to the stakeholder, the version mechanism also has the same problem. Stakeholder thinks that the version mechanism is complex because of the lack of documentation. Again this thesis has a generic introduction on the version mechanism that will mitigate the complexity of it.

For the visualizations, the stakeholder found the implemented visualizations very beneficial and user friendly. However, in general, the stakeholder has some concerns on design issues. For example, especially for the stacked bar chart, there are many feedbacks about the wrong usage of numbers, text and colors. Stakeholder thought that some of the conceptual visualizations are good, but have limitations. Besides the visualization, the stakeholder found the creation of the visualization with a significant effort. Although the stakeholder used the visualization configurator in order to create the visualization, he found it hard to use because of high availability of Tricia internal keywords. However, this thesis contains the documentation about the visualization configurator, which may decrease the complexity of it.

In the overall evaluation of Tricia, the stakeholder thought that their current system is mostly fulfils their requirements. However, when it comes to the version comparison, they had some limitations. According to them, they may use Tricia and their current system at the same time. For reporting and visualization for the present time, their current system will be used, but for the historical visualization, the stakeholder may use Tricia.

6.2. Presentation and Feedback

Other than the final interview, the presentation was given in the site of the stakeholder for the responsible team in order to get critical feedback about this thesis. The team consists of highly skilled international IT professionals. The presentation was prepared to them in order to introduce Tricia and the concept of this research. Feedbacks were generally concentrated on the design and the explanation of the visualization. As a general, they found the concept very beneficial, and they were very interested in it.

Their feedback about the master thesis is generally concentrated on the appearance of the visualizations. Generally, they found the concept of the visualization is proper and

enough for satisfying the needs. On the other hand, they recommended revising the selection of colors and the selection of words to explain the items of the visualization. For instance, especially in all of the visualizations, they told that change the color into the self-explanatory colors. In other words, they told that colors should have some meaning like red color represents failure and green color represents the success. Secondly, they recommended the usage of more straight words that do not need any other explanation. For instance, they told that the keyword called 'version' is very confusing. Instead of using 'version', they advised to use exact date of the times. Thirdly, they told that the visualizations of the whole IT-landscape are complicated. They prefer to see only the country specific visualizations. In other words, they request that omit the visualizations of the whole workspace because of their strain.

At last, these feedbacks are valuable since this team consists of very experienced international IT specialists. The feedbacks about the design issues will be corrected according to the literature [War04] for the very final presentation.

7. Future Work

This chapter contains information about the possible extensions and future research areas about this thesis work.

Firstly, the conceptual visualizations developed in the context of this thesis can be implemented in the future. These visualizations are radar plot and matrix representation. Besides this, other conceptual works can be updated in order to have more handy representation for the historical visualizations. Since there are many other visualizations in the literature that can be adapted to the visualization of historical EA models, the future work can be extended by having further literature review on visualizations.

Secondly, the visualization configurator should be more straightforward and simple. Some parts of the visualization configurator need expertise in order to create the visualizations. Since the new versions of Tricia have automatic pattern matching mechanism in the visualization configurator, this may decrease the complexity. However, it needs to be evaluated with the industry partners.

Currently the version mechanism does not allow users to store different versions inside the EA repository. The user creates the version each time by entering the desired date when there is a need. However, the old created versions can be stored inside the Tricia repository in order to create different visualization for the given version.

This thesis work does not include any visualization about forecasts and predictions (what-if analysis) because it only concentrated on the historical evaluation and visualization of EA models. However, this research is a starting point for the evaluation of EA models and future work may include the visualizations about the what-if analysis.

Part V.

Conclusion

8. Conclusion

To sum up, in the globalized world, successful companies are trying to adapt themselves to constantly changing market conditions. Because of the rapid development of the information technology, the organizations become more dynamic and world wide available, which also drives the complexity and high coupling structures at the same time. In order to cope with the given problems, EA Management started to be an important management discipline for the enterprises. Decision makers need to adapt their enterprises in a fast way by having efficient and precise decisions. Furthermore, they often need to evaluate their decisions by means of monitoring the evaluation of the enterprise.

The contribution of this research is to develop a mechanism, which aims to compare different versions of EA models and visualize them with understandable and user-friendly visualizations. The findings of the research evaluated in the German consumer goods industry with valuable feedbacks and future aspects.

In overall, this thesis work was successful to create several EA model states and retrieve them with understandable format by using EA management tool Tricia. Moreover, the understandable visualizations formed in order to compare the different EA models and resemble the evaluation of desired EA models. During the research, many different visualizations were found out both from the current practices in the industry and literature and mapped to the requirements to solve the limitations in the field historical visualization. The evaluation was accomplished by conducting the structured interview with the EA professionals and having presentations in the consumer goods industry. As general, the concept was evaluated as successful with the visualizations. After a decision of the concrete conceptual work, each step of it accomplished with the prototypical implementation. The implementation was evaluated in the consumer goods industry with valuable feedbacks. Since this is a new prototypical implementation, there is always a need for adding new functionalities and improving the concept (Chapter 7).

The starting point of this master thesis was the research questions, which determined the objectives of this work. To sum up, in order to achieve the goal of this thesis work, the following research questions were raised and answered.

- **What are the stakeholder requirements for visual EA model analysis and evaluation with respect to temporal aspects ?**

  The stakeholder often needs to compare different EA models, in order to assess their past decisions and to see the evaluation of their enterprise. According to stakeholder, these evaluations are needed to observe the current trends and changes in the company. This information is needed by the decision makers; firstly to take future decisions about the enterprise and secondly to detect any possible improvements of their organization and take tackles accordingly.

In order to get the requirements of the stakeholder, the initial interview was held. In the light of these requirements, it was realized that stakeholder eager to have an evaluation of their old decisions by means of examining the old EA model versions. The goals of the stakeholder were realized by having conceptual scenario that was explained detailed in Chapter 4.

- **What are the existing (EA) visualizations that can be realized to communicate temporal aspects of (EA) models?**

  There are many visualizations available both in the literature and the industry to visualize temporal data. However, there are not many sufficient and satisfying visualizations, which can be used to visualize EA model evaluation. After examining the current practices, this master thesis work accomplished the produce visualizations both conceptually and practically.

- **How to handle previous-states of an EA in a model repository and visualize them?**

  In order to handle the previous states, the version mechanism developed, which allows the administrator of the EA management tool to retrieve several versions according to the desired date. The version mechanism tracks all the changes on particular entity and has delta analysis to reach desired state. After successfully importing the enterprise data to the EA management tool (Tricia) and having different versions of EA models, the visualizations are generated by using the Tricia visualization configurator. After a long discussion with the stakeholder, the visualization that should be implemented had been determined. In this research, stacked bar chart, cluster map, layered cluster map and the interactive map visualizations were implemented to compare EA models. Moreover, tree structure visualizations, Spie Chart, FanLens, updated circle view, radar chart, Circos and matrix visualizations were conceptualized according to the requirements of the stakeholder. At the end, all of these visualizations were assessed by the stakeholder in the consumer goods industry.

Appendix

A. Appendix A

The following is the XML schema of the import document. It explains set of rules, structures that the XML document must obey. The import document for this research is based on the following XML schema. This schema is taken from the Tricia framework [1].

```
<?xml version="1.0" encoding="UTF-8"?>

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
        <xs:element name="mapping">
                <xs:complexType>
                        <xs:all>
                                <xs:element name="group" type="groupType" minOccurs="0"/>
                                <xs:element name="user" type="userType" minOccurs="0"/>
                                <xs:element name="membership" type="membershipType" minOccurs="0"/>
                                <xs:element name="spaces" minOccurs="0">
                                        <xs:complexType>
                                                <xs:sequence>
                                                        <xs:element name="space" type="spaceType" minOccurs="0"
                                                                maxOccurs="unbounded"/>
                                                </xs:sequence>
                                        </xs:complexType>
                                </xs:element>
                                <xs:element name="files" minOccurs="0">
                                        <xs:complexType>
                                                <xs:sequence>
                                                        <xs:element name="file" type="fileType" minOccurs="0" maxOccurs=
                                                                "unbounded"/>
                                                </xs:sequence>
                                        </xs:complexType>
                                </xs:element>
                                <xs:element name="relations" minOccurs="0">
                                        <xs:complexType>
                                                <xs:sequence>
                                                        <xs:element name="relation" type="relationType" minOccurs="0"
                                                                maxOccurs="unbounded"/>
                                                </xs:sequence>
                                        </xs:complexType>
                                </xs:element>
                                <xs:element name="tags" minOccurs="0">
                                        <xs:complexType>
                                                <xs:sequence>
                                                        <xs:element name="tag" type="tagType" minOccurs="0" maxOccurs="
                                                                unbounded"/>
                                                </xs:sequence>
                                        </xs:complexType>
                                </xs:element>
                        </xs:all>
                        <xs:attribute name="connectionprovider" type="xs:string" use="required"/>
                        <xs:attribute name="connectionurl" type="xs:string" use="required"/>
                        <xs:attribute name="connectionusername" type="xs:string" use="required"/>
                        <xs:attribute name="connectionpassword" type="xs:string" use="required"/>
                        <xs:attribute name="documentstoreroot" type="xs:string" use="optional"/>
                </xs:complexType>
        </xs:element>
</xs:schema>


        <xs:complexType name="spaceType">

                        <xs:all>
                                <xs:element name="readers" minOccurs="0">
                                        <xs:complexType>
                                                <xs:sequence>
                                                        <xs:element name="reader" type="principalType" minOccurs="0"
                                                                maxOccurs="unbounded"/>
                                                </xs:sequence>
                                        </xs:complexType>
                                </xs:element>
                                <xs:element name="writers" minOccurs="0">
                                        <xs:complexType>
```

[1] https://bitbucket.org/sascha.roth/tricia-visualizations-nevzat; visited on September 23th 2013.

```
                                        <xs:sequence>
                                                <xs:element name="writer" type="principalType" minOccurs="0"
                                                        maxOccurs="unbounded"/>
                                        </xs:sequence>
                                </xs:complexType>
                        </xs:element>
                        <xs:element name="pages" minOccurs="0">
                                <xs:complexType>
                                        <xs:sequence>
                                                <xs:element name="page" type="pageType" minOccurs="0" maxOccurs=
                                                        "unbounded"/>
                                        </xs:sequence>
                                </xs:complexType>
                        </xs:element>
                </xs:all>

                <xs:attribute name="name" type="xs:string" use="required"/>

        </xs:complexType>

<xs:complexType name="relationType">
                <xs:attribute name="tablename" type="xs:string" use="required"/>
                <xs:attribute name="tricianame" type="xs:string" use="required"/>
                <xs:attribute name="fromkeycolumnname" type="xs:string" use="required"/>
                <xs:attribute name="fromparent" type="xs:string" use="required"/>
                <xs:attribute name="tokeycolumnname" type="xs:string" use="required"/>
                <xs:attribute name="toparent" type="xs:string" use="required"/>
                <xs:attribute name="filter" type="xs:string" use="optional"/>
        </xs:complexType>

<xs:complexType name="tagType">
                <xs:attribute name="tablename" type="xs:string" use="required"/>
                <xs:attribute name="tricianame" type="xs:string" use="required"/>
                <xs:attribute name="idfield" type="xs:string" use="required"/>
                <xs:attribute name="tagfield" type="xs:string" use="required"/>
                <xs:attribute name="parent" type="xs:string" use="required"/>
                <xs:attribute name="filter" type="xs:string" use="optional"/>
        </xs:complexType>

        <xs:complexType name="pageType">

                        <xs:all>
                                <xs:element name="readers" minOccurs="0">
                                        <xs:complexType>
                                                <xs:sequence>
                                                        <xs:element name="reader" type="principalType" minOccurs="0"
                                                                maxOccurs="unbounded"/>
                                                </xs:sequence>
                                        </xs:complexType>
                                </xs:element>
                                <xs:element name="writers" minOccurs="0">
                                        <xs:complexType>
                                                <xs:sequence>
                                                        <xs:element name="writer" type="principalType" minOccurs="0"
                                                                maxOccurs="unbounded"/>
                                                </xs:sequence>
                                        </xs:complexType>
                                </xs:element>
                                <xs:element name="fields" minOccurs="0">
                                        <xs:complexType>
                                                <xs:sequence>
                                                        <xs:element name="field" type="fieldType" minOccurs="0"
                                                                maxOccurs="unbounded"/>
                                                </xs:sequence>
                                        </xs:complexType>
                                </xs:element>
                        </xs:all>
                <xs:attribute name="tablename" type="xs:string" use="required"/>
                <xs:attribute name="tricianame" type="xs:string" use="required"/>
                <xs:attribute name="namefield" type="xs:string" use="required"/>
                <xs:attribute name="idfield" type="xs:string" use="required"/>
                <xs:attribute name="filter" type="xs:string" use="optional"/>
                <xs:attribute name="contentfield" type="xs:string" use="optional"/>
                <xs:attribute name="lasteditorfield" type="xs:string" use="optional"/>
                <xs:attribute name="lastmodificationtimestampfield" type="xs:string" use="optional"/>
                <xs:attribute name="parentpagefield" type="xs:string" use="optional"/>
                <xs:attribute name="parentpageentitytype" type="xs:string" use="optional"/>
                <xs:attribute name="order" type="xs:string" use="optional"/>
                <xs:attribute name="idprefix" type="xs:string" use="optional"/>
                <xs:attribute name="usedefaultparentpage" type="xs:boolean" use="optional" default="false"/>
        </xs:complexType>

<xs:complexType name="fileType">

                        <xs:all>
                                <xs:element name="readers" minOccurs="0">
```

```
                                        <xs:complexType>
                                                <xs:sequence>
                                                        <xs:element name="reader" type="principalType" minOccurs="0"
                                                                maxOccurs="unbounded"/>
                                                </xs:sequence>
                                        </xs:complexType>
                                </xs:element>
                                <xs:element name="writers" minOccurs="0">
                                        <xs:complexType>
                                                <xs:sequence>
                                                        <xs:element name="writer" type="principalType" minOccurs="0"
                                                                maxOccurs="unbounded"/>
                                                </xs:sequence>
                                        </xs:complexType>
                                </xs:element>
                                <xs:element name="fields" minOccurs="0">
                                        <xs:complexType>
                                                <xs:sequence>
                                                        <xs:element name="field" type="fieldType" minOccurs="0"
                                                                maxOccurs="unbounded"/>
                                                </xs:sequence>
                                        </xs:complexType>
                                </xs:element>
                        </xs:all>
                <xs:attribute name="tablename" type="xs:string" use="required"/>
                <xs:attribute name="tricianame" type="xs:string" use="required"/>
                <xs:attribute name="namefield" type="xs:string" use="required"/>
                <xs:attribute name="idfield" type="xs:string" use="required"/>
                <xs:attribute name="filepathfield" type="xs:string" use="required"/>
                <xs:attribute name="directorynamelookupquery" type="xs:string" use="optional"/>
                <xs:attribute name="filter" type="xs:string" use="optional"/>
                <xs:attribute name="contentfield" type="xs:string" use="optional"/>
                <xs:attribute name="lasteditorfield" type="xs:string" use="optional"/>
                <xs:attribute name="lastmodificationtimestampfield" type="xs:string" use="optional"/>
                <xs:attribute name="parentpagefield" type="xs:string" use="optional"/>
                <xs:attribute name="parentpageentitytype" type="xs:string" use="optional"/>
                <xs:attribute name="order" type="xs:string" use="optional"/>
                <xs:attribute name="idprefix" type="xs:string" use="optional"/>

        </xs:complexType>


<xs:complexType name="principalType">
                <xs:attribute name="type" use="required">
                        <xs:simpleType>
                                <xs:restriction base="xs:string">
                                        <xs:enumeration value="groupname"/>
                                        <xs:enumeration value="username"/>
                                        <xs:enumeration value="groupid"/>
                                        <xs:enumeration value="userid"/>
                                        <xs:enumeration value="system"/>
                                </xs:restriction>
                        </xs:simpleType>
                </xs:attribute>
                <xs:attribute name="isderived" type="xs:boolean" use="optional" default="false"/>
                <xs:attribute name="value" type="xs:string" use="required"/>
        </xs:complexType>


        <xs:complexType name="fieldType">
                <xs:attribute name="columnname" type="xs:string" use="required"/>
                <xs:attribute name="tricianame" type="xs:string" use="required"/>
                <xs:attribute name="delimiter" type="xs:string" use="optional" default=";"/>
                <xs:attribute name="triciatype" use="optional" default="Text">
                        <xs:simpleType>
                                <xs:restriction base="xs:string">
                                        <xs:enumeration value="Text"/>
                                        <xs:enumeration value="Link"/>
                                        <xs:enumeration value="MultiText"/>
                                        <xs:enumeration value="Number"/>
                                        <xs:enumeration value="Date"/>
                                </xs:restriction>
                        </xs:simpleType>
                </xs:attribute>
                <xs:attribute name="parent" type="xs:string" use="optional"/>
                <xs:attribute name="filter" type="xs:string" use="optional"/>
                <xs:attribute name="importifnullorempty" type="xs:boolean" use="optional" default="false"/>
        </xs:complexType>


<xs:complexType name="groupType">
                <xs:attribute name="tablename" type="xs:string" use="required"/>
                <xs:attribute name="namefield" type="xs:string" use="required"/>
                <xs:attribute name="idfield" type="xs:string" use="required"/>
                <xs:attribute name="contentfield" type="xs:string" use="optional"/>
                <xs:attribute name="filter" type="xs:string" use="optional"/>
                <xs:attribute name="order" type="xs:string" use="optional"/>
                <xs:attribute name="idprefix" type="xs:string" use="optional"/>
        </xs:complexType>
```

```
<xs:complexType name="userType">
<xs:all>
                        <xs:element name="fields" minOccurs="0">
                                <xs:complexType>
                                        <xs:sequence>
                                                <xs:element name="field" type="fieldType" minOccurs="0"
                                                        maxOccurs="unbounded"/>
                                        </xs:sequence>
                                </xs:complexType>
                        </xs:element>
                </xs:all>
        <xs:attribute name="tablename" type="xs:string" use="required"/>
        <xs:attribute name="namefield" type="xs:string" use="required"/>
        <xs:attribute name="idfield" type="xs:string" use="required"/>
        <xs:attribute name="loginfield" type="xs:string" use="required"/>
        <xs:attribute name="passwordfield" type="xs:string" use="required"/>
        <xs:attribute name="contentfield" type="xs:string" use="optional"/>
        <xs:attribute name="filter" type="xs:string" use="optional"/>
        <xs:attribute name="order" type="xs:string" use="optional"/>
        <xs:attribute name="idprefix" type="xs:string" use="optional"/>
</xs:complexType>

<xs:complexType name="membershipType">
        <xs:attribute name="tablename" type="xs:string" use="required"/>
        <xs:attribute name="groupfield" type="xs:string" use="required"/>
        <xs:attribute name="userfield" type="xs:string" use="required"/>
        <xs:attribute name="groupfieldtype" use="optional" default="id">
                <xs:simpleType>
                        <xs:restriction base="xs:string">
                                <xs:enumeration value="name"/>
                                <xs:enumeration value="id"/>
                        </xs:restriction>
                </xs:simpleType>
        </xs:attribute>
        <xs:attribute name="userfieldtype" use="optional" default="id">
                <xs:simpleType>
                        <xs:restriction base="xs:string">
                                <xs:enumeration value="name"/>
                                <xs:enumeration value="id"/>
                        </xs:restriction>
                </xs:simpleType>
        </xs:attribute>
        <xs:attribute name="filter" type="xs:string" use="optional"/>
</xs:complexType>
```

Bibliography

[AGMS11] B. Alsallakh, E. Gröller, S. Miksch, and M. Suntinger. Contingency wheel: Visual analysis of large contingency tables. In *Proc. of the International Workshop on Visual Analytics (EuroVA 2011)*, pages 53–56. The Eurographics Association, The Eurographics Association, May 2011.

[AKK] M. Ankerst, D.A. Keim, and H.P. Kriegel. Circle segments: A technique for visually exploring large multidimensional data sets. In *In Visualization '96, Hot Topic Session*.

[AMM+08] W. Aigner, S. Miksch, W. MÃ¼ller, H. Schumann, and C. Tominski. Visual methods for analyzing time-oriented data. *Visualization and Computer Graphics, IEEE Transactions on*, 14(1):47–60, 2008.

[AR12] J.O. Aldrich and H.M. Rodriguez. *Building SPSS Graphs to Understand Data*, chapter 3. SAGE Publications, 2012.

[BBD] M. Burch, F. Beck, and S. Diehl. Timeline trees: visualizing sequences of transactions in information hierarchies. In 2008, editor, *Proceedings of the working conference on Advanced visual interfaces*, AVI '08, pages 75–82, New York, NY, USA. ACM.

[BDM+08] S. Buckl, T. Dierl, M. Matthes, R. Ramacher, and C.M. Schweda. Current and future tool support for ea management. *MDD, SOA und IT-Management*, 2008.

[BDMS10] S. Buckl, T. Dierl, F. Matthes, and C.M. Schweda. Building blocks for enterprise architecture management solutions. In Frank Harmsen, Erik Proper, Frank Schalkwijk, Joseph Barjis, and Sietse Overbeek, editors, *Practice-Driven Research on Enterprise Transformation*, volume 69 of *Lecture Notes in Business Information Processing*, pages 17–46. Springer Berlin Heidelberg, 2010.

[Ble11] R. Blewett. The importance of data visualization to business decision making. Technical report, Dundas Data Visualization, 2011.

[BMN10] T. Büchner, F. Matthes, and C. Neubert. Data model driven implementation of web cooperation systems with tricia. In *Proceedings of the Third international conference on Objects and databases*, ICOODB'10, pages 70–84, Berlin, Heidelberg, 2010. Springer-Verlag.

[BW08] L. Byron and M. Wattenberg. Stacked graphs – geometry & aesthetics. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1245–1252, November 2008.

[CCW12] J. Choy, V. Chawla, and L. Whitman. Data visualization techniques: From basics to big data with sas visual analytics. Technical report, SAS, 2012.

[Cha03] C. Chatfield. *The Analysis of Time Series: An Introduction, Sixth Edition*. Chapman & Hall/CRC Texts in Statistical Science. Taylor & Francis, 2003.

[Chi08] E.H. Chi. The social web: Research and opportunities. *IEEE Computer*, 41(9):88–91, 2008.

[CMS99] S.K. Card, J.D. Mackinlay, and B. Shneiderman, editors. *Readings in information visualization: using vision to think*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1999.

[Coo08] N. Cook. *Enterprise 2.0: How Social Software Will Change the Future of Work*. Gower Publishing Ltd, 2008.

[ctv13] Circos table viewer. `http://mkweb.bcgsc.ca/tableviewer/`, 2013.

[ELSW06] A.M. Ernst, J. Lankes, C.M. Schweda, and A. Wittenburg. Tool support for enterprise architecture management - strengths and weaknesses. In *EDOC*, pages 13–22. IEEE Computer Society, 2006.

[Fei03] D.G. Feitelson. Comparing partitions with charts, 2003.

[Fri94] M. Friendly. A fourfold display for 2 by 2 by k tables. Technical Report 217, York University, Psychology Dept, 1994.

[Fri05] M. Friendly. Milestones in the history of data visualization: A case study in statistical historiography. In C. Weihs and W. Gaul, editors, *Classification: The Ubiquitous Challenge*, pages 34–52. Springer, New York, 2005.

[HBO10] J. Heer, M. Bostock, and V. Ogievetsky. A tour through the visualization zoo. *Commun. ACM*, 53(6):59–67, June 2010.

[HHWN02] S. Havre, E. Hetzler, P. Whitney, and L. Nowell. Themeriver: visualizing thematic changes in large document collections. *Visualization and Computer Graphics, IEEE Transactions on*, 8(1):9–20, 2002.

[HK07] M.J. Halvey and M.T. Keane. An assessment of tag presentation techniques. In *WWW '07: Proceedings of the 16th international conference on World Wide Web*, pages 1313–1314, New York, NY, USA, 2007. ACM Press.

[HKA09] J. Heer, N. Kong, and M. Agrawala. Sizing the horizon: the effects of chart size and layering on the graphical perception of time series visualizations. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1303–1312, New York, NY, USA, 2009. ACM.

[HMRS] M. Hauder, F. Matthes, S. Roth, and C Schulz. Generating dynamic cross-organizational process visualizations through abstract view model pattern matching. In *Architecture Modeling for Future Internet enabled Enterprise*.

[ID90] A. Inselberg and B. Dimsdale. Parallel coordinates: a tool for visualizing multi-dimensional geometry. In *Visualization, 1990. Visualization '90., Proceedings of the First IEEE Conference on*, pages 361–378, 1990.

[JLtD+06] H. Jonkers, M. Lankhorst, H.L. ter Doest, F. Arbab, H. Bosma, and R.J. Wieringa. Enterprise architecture: Management tool and blueprint for the organisation. *Information Systems Frontiers*, 8(2):63–66, 2006.

[JS91] B. Johnson and B. Shneiderman. Tree-maps: a space-filling approach to the visualization of hierarchical information structures. In *Visualization, 1991. Visualization '91, Proceedings., IEEE Conference on*, pages 284–291, 1991.

[Kan00] E. Kandogan. Star coordinates: A multi-dimensional visualization technique with uniform treatment of dimensions. In *In Proceedings of the IEEE Information Visualization Symposium, Late Breaking Hot Topics*, pages 9–12, 2000.

[KBS04] D. Krafzig, K. Banke, and D. Slama. *Enterprise SOA: Service-Oriented Architecture Best Practices (The Coad Series)*, pages 1–12. Prentice Hall PTR, Upper Saddle River, NJ, USA, 2004.

[Kei02] D.A. Keim. Information visualization and visual data mining. *IEEE Transactions on Visualization and Computer Graphics*, 8(1):1–8, January 2002.

[KHDH02] D.A. Keim, M.C. Hao, U. Dayal, and M. Hsu. Pixel bar charts: a visualization technique for very large multi-attribute data sets. *Information Visualization*, 1(1):20–34, March 2002.

[KSB+09] M.I. Krzywinski, J.E. Schein, I. Birol, J. Connors, R. Gascoyne, D. Horsman, S.J. Jones, and M.A. Marra. Circos: An information aesthetic for comparative genomics. *Genome Research*, 2009.

[KSS04] D.A. Keim, J. Schneidewind, and M. Sips. Circleview: a new approach for visualizing time-related multidimensional data set. In *Proceedings of the working conference on Advanced visual interfaces*, AVI '04, pages 179–182. ACM, 2004.

[Lee99] Y.T. Lee. Information modeling: From design to implementation. In *Proceedings of the Second World Manufacturing Congress*, pages 315–321, 1999.

[Lew06] P.J. Lewi. *Speaking of Graphics*, chapter 5. February 2006.

[LRKC10] B. Lee, N.H. Riche, A.K. Karlson, and S. Carpendale. Sparkclouds: Visualizing trends in tag clouds. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1182–1189, November 2010.

[MÖ8] F. Mörchen. Organic pie charts. In *ICDM*, pages 947–952. IEEE Computer Society, 2008.

[McA06] A.P. McAfee. Enterprise 2.0 - the down of emergent collaboration. *MIT Sloan Management Review*, 47(3), 2006.

[MER00] C.J. Morris, D.S. Ebert, and P. Rheingans. An experimental analysis of the effectiveness of features in chernoff faces. In *28th AIPR Workshop: 3D Visualization for Data Exploration and Decision Making, Proceedings of SPIE*, pages 12–17, 2000.

[mic13] `http://office.microsoft.com/de-at/word-help/erstellen-einer-smartart-grafik-HA001205867.aspx`, 2013. Accessed: 2013-09-10 06:46:11 +0200.

[MM99] H. Mosley and A. Mayer. Benchmarking national labour market performance: a radar chart approach. Discussion papers, research unit: Labor market policy and employment, Social Science Research Center Berlin (WZB), 1999.

[MM12] F. Matthes and N. Matthes. Praxiserfahrungen beim einsatz von hybriden wikis fÃ¼r das agile management von it-architekturinformationen, 2012.

[MMN88] J.E. McDonald, M.E. Molander, and R.W. Noel. Color-coding categories in menus. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '88, pages 101–106, New York, NY, USA, 1988. ACM.

[MN11] F. Matthes and C. Neubert. Enabling knowledge workers to collaboratively add structure to enterprise wikis. In *12th European Conference on Knowledge Management - ECKM 2011*, University of Passau, Germany, 2011.

[MNS11] F. Matthes, C. Neubert, and A. Steinhoff. Hybrid wikis: Empowering users to collaboratively structure information. In María José Escalona Cuaresma, Boris Shishkov, and José Cordeiro, editors, *ICSOFT (1)*, pages 250–259. SciTePress, 2011.

[MNS13] F. Matthes, C. Neubert, and A. Steinhoff. Facilitating structuring of information for business users with hybrid wikis. In MaríaJosé Escalona, José Cordeiro, and Boris Shishkov, editors, *Software and Data Technologies*, volume 303 of *Communications in Computer and Information Science*, pages 237–251. Springer Berlin Heidelberg, 2013.

[Neu12] C. Neubert. *Facilitating Emergent and Adaptive Information Structures in Enterprise 2.0 Platforms*. PhD thesis, Fakultät für Informatik der Technischen Universität, 2012.

[Ngu12a] V.X. Nguyen. Circosonic: a sonification of circos, a circular graph of pair wise table data. In *Proceedings of the International Conference on Auditory Display (ICAD)*, 2012.

[Ngu12b] V.X. Nguyen. Circosonic: a sonification of circos, a circular graph of pair wise table data. In *The International Community for Auditory Display, 2012*. Georgia Institute of Technology, 2012.

[O'R05] T. O'Reilly. What is web 2.0: Design patterns and business models for the next generation of software. 2005.

[ORA13] ORACLE. Oracle learning library. `http://www.oracle.com/technetwork/tutorials/index.html`, 2013.

[QZP03] C. Qin, C. Zhou, and T. Pei. Taxonomy of visualization techniques and systems - concerns between users and developers are different. In *Asia GIS Conference 2003*, 2003.

[RC] R. Rao and S.K. Card. The table lens: merging graphical and symbolic representations in an interactive focus + context visualization for tabular information. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '94, pages 318–322, New York, NY, USA. ACM.

[RM13] S. Roth and F. Matthes. Future research topics in enterprise architectures evolution analysis. In *Design for Future*, 2013.

[RMC91] G.G. Robertson, J.D. Mackinlay, and S.K. Card. Cone trees: animated 3d visualizations of hierarchical information. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '91, pages 189–194, New York, NY, USA, 1991. ACM.

[Rob12] N. Robbins. *A Histogram is NOT a Bar Chart*. 2012.

[RR06] S. Robertson and J.C. Robertson. *Mastering the Requirements Process*, chapter Appendix B. ACM Press Books. Addison Wesley Professional, 2006.

[SBB+03] N. Sheth, K. Börner, J. Baumgartner, K. Mane, and Eric A. Wernert. Treemap, radial tree and 3d tree visualizations. In *IEEE Information Visualization Conference Poster Compendium*, pages 128–129, 2003.

[SCL+12] Conglei Shi, Weiwei Cui, Shixia Liu, Panpan Xu, Wei Chen 0001, and Huamin Qu. Rankexplorer: Visualization of ranking changes in large time series data. *IEEE Trans. Vis. Comput. Graph.*, 18(12):2669–2678, 2012.

[Sel10] H.J. Seltman. Experimental design and analysis, 2010.

[SGW05] S. Schaffert, A. Gruber, and R. Westenthaler. A semantic wiki for collaborative knowledge formation. In *Proceedings of SEMANTICS 2005 Conference.*, 2005.

[Shn96] B. Shneiderman. The eyes have it: a task by data type taxonomy for information visualizations. In *Visual Languages, 1996. Proceedings., IEEE Symposium on*, pages 336–343, 1996.

[SLF+11] Massimo Stafoggia, Adele Lallo, Danilo Fusco, Anna P. Barone, Mariangela D'Ovidio, Chiara Sorge, and Carlo A. Perucci. Spie charts, target plots, and radar plots for displaying comparative outcomes of health care. *Journal of clinical epidemiology*, January 2011.

[SLKS12] H. Song, B. Lee, B. Kim, and J. Seo. Diffmatrix: Matrix-based interactive visualization for comparing temporal trends. In *Eurographics Conference on Visualization (EuroVis)*, pages 103–107, 2012.

[SMR12] M. Schaub, F. Matthes, and S. Roth. Towards a conceptual framework for interactive enterprise architecture management visualizations. In Elmar J. Sinz and Andy SchÃ¼rr, editors, *Modellierung*, volume 201 of *LNI*, pages 75–90. GI, 2012.

[Som07a] I. Sommerville. *Software Engineering*, page 152. International computer science series. Addison-Wesley, 2007.

[Som07b] I. Sommerville. *Software Engineering*, pages 119–121. International computer science series. Addison-Wesley, 2007.

[Som07c] I. Sommerville. *Software Engineering*, pages 409–412. International computer science series. Addison-Wesley, 2007.

[Spe07a] R. Spence. *Information Visualization: Design for Interaction (2nd Edition)*, chapter Chapter 1. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 2007.

[Spe07b] R. Spence. *Information Visualization: Design for Interaction (2nd Edition)*, pages 89–90. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 2007.

[SS11] A.W. Schneider and A. Steinhoff. Applying web analytics tools in the context of enterprise social software. In *Proceedings of the 12th European Conference on Knowledge Management (ECKM)*, Passau, Germany, 2011.

[Sto10] A. Stobbe. Enterprise 2.0: How companies are tapping the benefits of web 2.0. Technical report, Deutsche Bank Research, Deutsche Bank Research Frankfurt am Main Germany, September 2010.

[TN09] BÃ¼chner T. and C. Neubert. A concept and service based analysis of commercial and open source enterprise 2.0 tools. In *International Conference on Knowledge Management and Information Sharing*, Madeira, Portugal, 2009.

[TR09] S. Thakur and T.M. Rhyne. Data vases: 2d and 3d plots for visualizing multiple time series. In *Advances in Visual Computing*, volume 5876 of *Lecture Notes in Computer Science*, pages 929–938. Springer Berlin Heidelberg, 2009.

[Tuf86a] E.R. Tufte. *The visual display of quantitative information*, page 24. Graphics Press, Cheshire, CT, USA, 1986.

[Tuf86b] E.R. Tufte. *The visual display of quantitative information*, pages 40–42. Graphics Press, Cheshire, CT, USA, 1986.

[Tuf97] E.R. Tufte. *Visual Explanations: Images and Quantities, Evidence and Narrative*. Graphics Press, 1997.

[TW10] V. Tirronen and M. Weber. Sparkline histograms for comparing evolutionary optimization methods. In Joaquim Filipe and Janusz Kacprzyk, editors, *IJCCI (ICEC)*, pages 269–274. SciTePress, 2010.

[Vin07] L. Vincent. Taking online maps down to street level. *Computer*, 40(12):118–120, 2007.

[vol13] http://www.volere.co.uk/template.htm, 2013. Accessed: 2013-09-10 06:30:06 +0200.

[War04] C. Ware. *Information Visualization: Perception for Design*. Interactive Technologies. Elsevier Science, 2004.

[WBL+05] A. Wegmann, P. Balabko, L. LÃª, G. Regev, and I. Rychkova. A method and tool for business-it alignment in enterprise architecture. In Orlando Belo, Johann Eder, JoÃ£o FalcÃ£o e Cunha, and Oscar Pastor, editors, *CAiSE Short Paper Proceedings*, volume 161 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2005.

[web13a] http://www.infoasset.de/, 2013. Accessed: 2013-09-13 15:32:08 +0200.

[web13b] http://wwwmatthes.in.tum.de/pages/t5ma0jrv6q7k/sebis-Public-Website-Home, 2013. Accessed: 2013-09-13 15:34:15 +0200.

[WGK10] M. Ward, G. Grinstein, and D. Keim. *Interactive Data Visualization: Foundations, Techniques, and Applications*, pages 271–278. A. K. Peters, Ltd., Natick, MA, USA, 2010.

[Wil09] G.J. Wills. Visualizing hierarchical data. In *Encyclopedia of Database Systems*, pages 3425–3432. Springer US, 2009.

[XST08] L. Xinghua, L. Shixia, and W. Tianshu. Fanlens: A visual toolkit for dynamically exploring the distribution of hierarchical attributes. In *Visualization Symposium, 2008. PacificVIS '08. IEEE Pacific*, pages 151–158, 2008.

[YVMF06] F.W. Young, P. Valero-Mora, and M. Friendly. *Visual statistics: seeing data with dynamic interactive graphics*, pages 19–20. Wiley-Interscience, 2006.

[ZDFB12] J. Zhao, S.M. Drucker, D. Fisher, and D. Brinkman. Timeslice: interactive faceted browsing of timeline data. In *Proceedings of the International Working Conference on Advanced Visual Interfaces*, AVI '12, pages 433–436, New York, NY, USA, 2012. ACM.

[Zol07] A. Zollers. Emerging motivations for tagging: Expression, performance, and activism. In *Tagging and Metadata for Social Information Organization Workshop, WWW07*, 2007.