REQUIREMENTS FOR AUTOMATED ENTERPRISE ARCHITECTURE MODEL MAINTENANCE

A Requirements Analysis based on a Literature Review and an Exploratory Survey

Matthias Farwick, Berthold Agreiter, Ruth Breu
Institute of Computer Science, University of Innsbruck, Technikerstr. 21A, 6020 Innsbruck, Austria

Steffen Ryll, Karsten Voges, Inge Hanschke
iteratec GmbH, Inselkammerstr. 4, 82008 Unterhaching, Germany

Keywords: EAM, Enterprise architecture management, Model maintenance, Enterprise information integration, Model evolution.

Abstract: Enterprise Architecture Management (EAM) is the practice of modeling the business and IT artifacts in an enterprise and relating them with each other. By documenting these interdependencies between business and the supporting IT, strategic decisions can be made towards a planned and consolidated enterprise architecture that matches the business needs. However, enterprise architecture models can grow very large, thus making the manual creation and maintenance of these models a difficult and time consuming task. From this, we derived the three research questions of this paper: (i) Which related work exists on the maintenance of EA models, and does it refer to automation techniques? (ii) Is EA maintenance automation demanded by practitioners? If yes, (iii) What are the requirements for such an automation method and tool? In this paper we tackle these questions by conducting a literature analysis on EA literature from research and practice. In addition, we present the results of a survey among EA practitioners we conducted to find out current maintenance practices. We then describe a collection of requirements for an automated EA maintenance method and tool, that we derived from the results of the survey, the findings in the literature review and our own experience as EA consultants. Finally, we present several success evaluation criteria for an automated EA maintenance solution.

1 INTRODUCTION

Enterprise Architecture Management (EAM) is a practice used in mid-sized to large organizations that documents the relationships between business, its supporting software and the underlying IT infrastructure. This effort is commonly made for the goals of better aligning business and IT, enabling strategic planning, to assess risks, and to check compliance with legal regulations. EAM is also introduced in many companies to enable better communication between the different stakeholders. Several enterprise architecture (EA) frameworks have been proposed and are in use today, such as The Open Group Architecture Framework (TOGAF) (The Open Group, 2009), or the Zachman Framework (Zachman, 1987). These frameworks commonly describe a metamodel used to describe the EA data, organizational best practices and governance mechanisms.

In the practice of enterprise architecture, the corresponding EA models can grow very large and can expose complex relationships between EA model elements. Also, the large and complex models continuously need to be aligned with the real–world enterprise which they are supposed to represent. For example mergers & acquisitions, new business strategies, changing market requirements, or the exploration of new markets can immensely change the business and IT architecture of an enterprise. This combination of complex and evolving models means that a large effort needs to be put into keeping EA models in–sync with what they represent in the real world.

Many recent publications from research (Fischer et al., 2007; Hafner and Winter, 2008) and practice (Hanschke, 2009; The Open Group, 2009; US Department of Defense, 2010), have focused on the EA management processes that need to be implemented in order to keep EA models up-to-date and increase

Farwick M., Agreiter B., Breu R., Ryll S., Voges K. and Hanschke I..
REQUIREMENTS FOR AUTOMATED ENTERPRISE ARCHITECTURE MODEL MAINTENANCE - A Requirements Analysis based on a Literature Review and an Exploratory Survey.
DOI: 10.5220/0003429203250337
In *Proceedings of the 13th International Conference on Enterprise Information Systems* (ICEIS-2011), pages 325-337
ISBN: 978-989-8425-56-0

the return on investment (ROI) for EA efforts. In fact, there exists evidence that organizational and people-oriented strategies are the key success factors for EA initiatives (Ambler, 2010). This explains why most EA literature has so far only addressed the optimization of organizational EA strategies.

However, the evolutionary characteristic of EA models, that forces EA practitioners to continuously update the models manually, has yet only been addressed via these management strategies, hence the problem of cumbersome manual EA model maintenance persists. No publications could be identified that actually looks at automated tool-aided strategies to keep EA models up-to-date. Therefore, we conducted a survey to find out whether automation mechanisms are actually demanded in practice, and if yes, which kinds of features are desired for such a system. In this paper we collect the findings of the survey, related work and our own experience in the field, to compile a set of requirements for a tool that aids automated EA maintenance. These requirements can then later on be used to create technical solutions to reduce the manual effort of EA maintenance.

The paper is organized as follows: the following section outlines the motivation for conducting this work and describes the contribution. In Section 3 relevant related work on EA and automated information retrieval is presented. After that, Section 4 describes the survey setup and survey responses about how enterprise architecture management is currently done in practice, and in particular where the practitioners see room for improvement. The discussion of survey responses continues in Section 5 where a set of requirements for automated EA model maintenance is developed by referring to the responses and previously introduced related work. To be able to determine whether a future tool for automated model maintenance fulfills the expectations of the respective stakeholders, Section 6 sketches success evaluation criteria. Finally, Section 7 concludes and gives an outlook on future work.

2 MOTIVATION & CONTRIBUTION

According to (Winter et al., 2010) and (Buckl et al., 2009b) the majority of EA practitioners rely on the manual input of changes to an EA model, without any automation of EA model updates. While it is realistic to assume that manual data input will be the main source of changes to EA repositories in the foreseeable future, we still identified significant room for improvements in this area, from our experience in the field, and discussions with experts.

We argue that the next step to increasing productivity, efficiency and finally ROI of EA initiatives can be reached by automating the maintenance processes for EA models. By automated maintenance we refer to the automated update, or addition, of model elements in a EAM tool, with minimal human intervention.

To find out whether this kind of automated maintenance is demanded, and which type of information sources are appropriate, we conducted an online survey among practitioners.

The contribution of this paper is twofold. First, we give an overview of literature that supports the above claim that EA maintenance needs to be improved. We further confirm it by the results of the survey. Second, we derive high-level requirements for a tool supporting automated EA maintenance by using the results of the survey, related literature and our own experience as EA consultants with the company iteratec GmbH. In addition we describe high-level evaluation criteria that can be used to evaluate a system that implements the defined requirements.

3 BACKGROUND & RELATED WORK

In ANSI/IEEE 1471 (IEEE Computer Society, 2000) the term “architecture” is defined as *“The fundamental organization of a system, embodied in its components, their relationships to each other and the environment, and the principles governing its design and evolution”*. Thus, enterprise architecture management is the practice of relating and documenting the business and IT components of an organization. The publication by Winter and Fischer (Winter and Fischer, 2007) gives an overview of the common artifacts and layers of the models that are often created in the course of EA initiatives. An important characteristic of EA is that not only models of the current architecture are created, but also of future to-be architectures. These to-be architectures are the goal to which the sum of all transformations of the enterprise architecture should ultimately lead. The reasons why EAM initiatives are started in companies are manifold. One of the most typical reasons is its capability of fostering business/IT alignment. However, there exist many other reasons why EAM is applied, such as architecture documentation, standardization, and ensuring regulatory compliance (Winter et al., 2010).

In the following we introduce enterprise architecture frameworks and discuss how they relate to EA model maintenance. After that, we look at research publications that focus on EA maintenance management processes to find out whether they are capable to support automated EA maintenance. We then discuss related work in the field of automated systems operations maintenance and EA model change propagation. Finally, we discuss which automated EA maintenance features are exposed by current EA tools.

Enterprise Architecture Frameworks. Since one of the first comprehensive EA frameworks, the Zachmann framework (Zachman, 1987), was published in 1987, many different EA frameworks for differing domains have been proposed, refined and put into practice. Among these are the frameworks with a governmental background such as Department of Defence Architecture Framework (US Department of Defense, 2010) and the Federal Enterprise Architecture Framework (Chief Information Officers Council, 2007), as well as frameworks such as TOGAF (The Open Group, 2009) which are developed independently from governmental bodies. In addition, there exist different EA modeling notations such as ArchiMate (Lankhorst, 2005) and UML profiles. In general, such EA frameworks prescribe the type of artifacts to collect in an EA model, different views on these models, and organisational aspects of EA initiatives. However, these frameworks fall short of concrete advice for how to collect EA data and maintain its quality attributes actuality, consistency and accuracy (Buckl et al., 2010). For example, the Architecture Development Method (ADM) embedded in TOGAF simply states that some kind of "monitoring" should be applied to keep the models up-to-date, but does not state what this implies in practice.

EA Management Processes. Since we want to identify related work in automated EA maintenance, we now look at EA management publications that investigate the maintenance of EA models. There exist many publications in the area of EA management methods and organizational strategies. However, little work was found that actually discussed the automated maintenance of EA repositories. Or as Buckl et al. put it in their recent literature analysis: *"Identifying, gathering, and maintaining knowledge about the EA is a challenge, which is only recently addressed by isolated approaches ... the analyzed EA management methods do not detail on how to acquire and incorporate knowledge from other sources"* (Buckl et al., 2010).

Several publications recognize that EA model maintenance is challenging due to its evolutionary character such as (Hafner and Winter, 2008) and (Buckl et al., 2010), but do not further investigate on how to improve the current methods via automation.

Buckl et al. conducted a survey to elicit the current state of EA practice in organizations. They report a high interest of their survey participants for automated EA information gathering (Buckl et al., 2009b). In particular the respondents showed interest in the automated integration of information from Configuration Management Databases (CMDBs) into their EA repositories. Again, no follow up publications could be found on these findings.

In another publication by Buckl et al. the authors acknowledge the evolutionary characteristic of EA and describe a meta-model that incorporates temporal information into the models (Buckl et al., 2009a). According to them this enables that time-based reevaluation events can be fired, and enables to check whether projects are on time.

Only very few publications were identified that propose a concrete maintenance process for EA, which could possibly be automated. Among these is the paper by Fischer et al. (Fischer et al., 2007) who propose an EA maintenance process in which all information providers to EA (e.g. departments) are responsible for maintaining their EA data themselves and for providing it to a federated EA repository. They argue that by this federated approach, the data quality is kept high since each stakeholder can work with the kind of management tool he is familiar with. However, the authors state that the process has not been implemented on an automated level and no concrete requirements have been identified for an implementation. Another publication which proposes an automated maintenance process is the work by Moser et al. (Moser et al., 2009). Their paper discusses several EA process patterns which, according to them, are recurring best practices they apply in organizations they consult. One of these patterns is called "Automatic Data Acquisition/Maintenance". They also state that an *"adequate EAM tool needs to be in place"*, but do not state exact requirements for such a tool. This further confirms the research direction of our work. Hafner & Winter propose an enterprise application architecture management process (Hafner and Winter, 2008). However, they do not discuss the possible automation of this process. In (Farwick et al., 2010) we describe a method and process to integrate infrastructure cloud instances into an enterprise architecture tool. There we focused on the specific requirements for integrating cloud interfaces, whereas in the current publication we concentrate on a generic solution to automated EA maintenance.

Automated Configuration Management. A number of recent publications have looked at the problem of automating updates of configuration data in Configuration Management Databases (CMDB) (OGC, 2007). Although, the granularity of the data kept in such repositories is more fine-grained than typical EA information items, these approaches are relevant from a technical point of view. Konstantinou and Yemini describe a distributed configuration management data layer that is based on a peer-to-peer architecture (Konstantinou and Yemini, 2009). A different approach is followed by Tanner et al. who propose a semantic integration strategy to collect management data (Tanner and Feridun, 2009). An interesting development is also the standardization effort by the Distributed Management Task Force (DMTF), called Configuration Management Database Federation (CMDBf) (DMTF, 2010). This specification proposes standard interfaces and a reference architecture for the exchange of systems operation management data. However, it is yet unclear whether the specification will find wide-spread adoption among EA tool and enterprise information system vendors.

EA Change Propagation. Recently the problem of change propagation in EA models was discussed in several papers. De Boer et al. conceptually describe a method for calculating change impact within the ArchiMate EA modeling language (De Boer, F.S. et al., 2005). A more sophisticated approach is presented by Dam et al. (Dam et al., 2010) who propose a logic-based approach to automatically infer secondary changes, after an initial change to a model, that need to be performed in order to maintain consistency of an EA model. In another publication on change in EA models Kumar et al. (Kumar et al., 2008) create an enterprise interaction ontology to calculate impacts of changes in EA models. Although, both are promising approaches to change propagation and change impact analysis, the papers do not discuss how the initial changes can be detected in an automated way.

Current EA Tool Landscape. The landscape of Enterprise Architecture Management tool support has grown and matured in the last decade. It ranges from large scale commercial tools like Troux[1], to open-source EA applications like the Essential Project[2]. Seven years ago ter Doest and Lankhorst stated in (ter Doest and Lankhorst, 2004):
"We expect that in 7 years from now, enterprise architecture will be a real-time tool for management and redesign of the enterprise for better performance, flexibility and agility. (...) There will no longer (be) a boundary between the descriptive nature of architecture models and the operational side comprising a.o. business process management and IT management.". However, today this vision has not yet materialized. Current EA tools are restricted to import and export of specific exchange formats such as XMI or Excel. Also, vendors tend to create specific integrations with external tools from either the own tool suite, or selected partner companies. The current Gartner Magic Quadrant for Enterprise Architecture Tools gives an overview of the capabilities of current commercial EA tools (Wilson, 2010).

4 SURVEY SETUP & GENERAL FINDINGS

The discussion of related work in the previous section shows that there is little literature that is specifically targets automated EA maintenance. A reason for this could potentially be that such features are not required in practice. Thus, one of the reasons why we conducted the survey, was to find out whether EA automation *is* demanded in practice and whether practitioners are satisfied with their current EA maintenance processes and tool support. Secondly, we wanted to find out how EA maintenance is actually conducted in companies, and which tool support is demanded to aid EA maintenance. The findings of this second part are described in Section 5.

Our survey was conducted as an online poll with 34 questions and was created in collaboration with the EAM tool vendor iteratec. Survey participants were personally invited via e-mails, and calls for participation were posted in EA-focused online communities.

The survey consisted of several general questions regarding the industry sector and the country of operation, as well as questions regarding the respondents' confidence with EA topics. In addition, it asked for the years of experience of the respondents and whether the company of the respondent currently applies an EA initiative. This allowed to raise the quality of the results by eliminating respondents which had little EA experience or did not currently work in the field. With the remainder of the questions we tried to elicit current EA practices and the requirements that the practitioners envision for automated EA maintenance.

The survey was open for 25 days and 43 completed responses were gathered. From these responses 11 participants reported that their company is currently not involved in an EA project and additional 3

[1] http://www.troux.com last retrieved 1/22/11

[2] http://www.enterprise-architecture.org last retrieved 1/22/11

respondents reported that they only have little confidence in EA topics. These responses were discarded, leading to a sample size of 29 responses (n=29). The language of the survey was English to avoid that only persons from a specific geographic area could participate. However, it is not possible to state how well the respondents represent the set of all EA practitioners.

4.1 General Findings

Demographics. The majority of the survey participants resides in Europe, with Germany having the highest participation rate of 38%. Another 17% of the participants reside in the USA, 14% in the United Kingdom, and the remainder was distributed over several European and Asian countries.

The industry sector of survey the participants was fairly distributed with the Finance/Insurance/Banking sector being represented the most by 28% of the participants (see Figure 1).

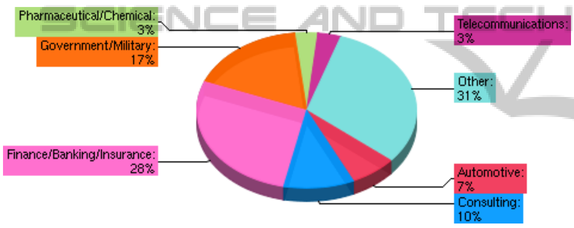

Figure 1: The industry sectors of the survey participants.

Current EA Processes. 90% of the survey participants agreed or strongly agreed to the statement “The manual collection and maintenance of EA data in a sufficient quality, is one of the major challenges of EA practice.” (Figure 2). Also, 55.2% state that the amount of manual EA data collection is “barely acceptable” (41.4%) or even “inacceptable” (13.8%). This clearly highlights that there is a demand in practice to optimize the data collection methods and processes for EA projects and shows that the current situation is not satisfactory to many.

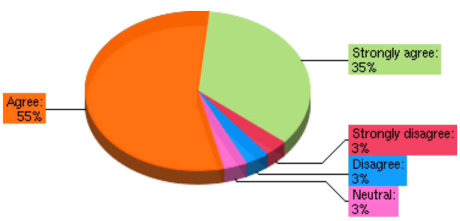

Figure 2: Results for the question: *“Do you agree with the following sentence: The manual collection and maintenance of EA data in a sufficient quality, is one of the major challenges of EA practice.”?*

To gather more insights on the goals of the participating EA practitioners, we asked them to sort a list of goals according to their priorities. These are the results[3]:

1. Business to IT Alignment (Score 133).
2. Strategic Planning (Roadmapping) (Score 120).
3. Architecture Documentation (Score 88).
4. Identification of Possible Savings (Score 88).
5. Decision Support (Score 83).
6. Regulatory Compliance (Score 66).

Furthermore, out of all respondents, 45% reported that their EA repository is maintained by the EA responsibles in a continuous manner. 35% declared that their repositories are updated in recurring efforts, and 21% indicated that they use a different scheme. Of those who update in recurring efforts, 50% reported that their repository is updated every two months or in even longer intervals. The other 50% stated that the repository is updated on demand. These findings indicate that almost one fifth (17.5%) of the participants rely their decisions on data which is potentially as old as two months. We also asked the question of what level of actuality is desired for the EA goals in the organization (see Figure 3). 48% of the respondents stated that the actuality within days or up to four weeks is appropriate for their purpose. However, out of these 48%, 42.8% were from the group that indicated update cycles that are longer than two months (roughly 20% of all participants). This possibly indicates that there is a disparity between the desired and the effective actuality of data in some organizations. Further investigation is, however, needed to confirm this finding.

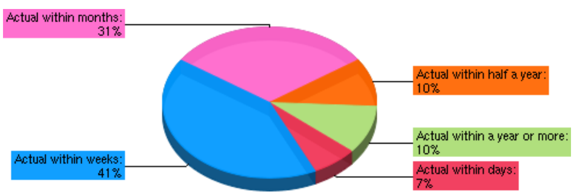

Figure 3: Desired actuality of EA data for the goals of the respondents.

One of our primary interests was to find out whether automation techniques are actually applied in the current EA practice of the respondents. 20.7% actually indicated that they are using automated mechanisms to update their EA repository. However, no re-

[3]Score is a weighted calculation. Items ranked first are valued higher than the following ranks, the score is the sum of all weighted rank counts.

spondent further described this automation in the optional clarification text field.

Current EA Model Element Coverage and Desired Coverage. In a different set of questions, we wanted to elicit which type of collected data in an EA repository exhibits a mismatch between the percentage of actually collected EA elements and the desired percentage to be collected. Furthermore, we wanted to obtain a quantification of how big the gap among them is. Here, we asked, for example, what percentage of infrastructure elements is currently covered in the EA repository, and what percentage would be considered as optimal for the EA goals of the enterprise. Figure 4 shows the average over the responses for the element types *infrastructure element*, *project*, and *information system*.

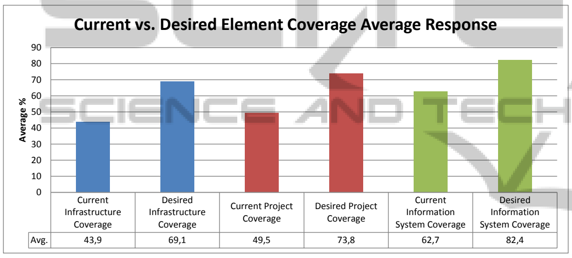

Figure 4: Current vs. desired EA element coverage.

An obvious pattern that can be seen in the chart is that the respondents are not satisfied with the coverage of the respective EA information items, i.e. the desired coverage is always higher than the actual coverage. The demand for a high coverage of information systems is the highest with 82.4%. Desired coverage for infrastructure is the lowest on average, however the discrepancy between the current and desired state is the largest. The reason for this could be that it is more difficult to acquire a complete picture of the IT infrastructure, than a complete picture of existing information systems in an organization. Again, automation methods might be able to aid this situation.

Desired Features & Information Sources. With another set of questions we wanted to evaluate the subjective usefulness that the respondents see in specific automation solutions. Figure 5 gives an overview of the results. The participants were asked to rank the usefulness of a specific feature on a Likert scale from 1 to 5, where 1 corresponded to "Not useful" and 5 corresponded to "Very useful". Similar to the chart of Figure 4, the detection of information system-related data is regarded to have the highest usefulness by the participants. The automated detection of project information has the lowest outcome in this evaluation. A reason for this might be that information about projects is relatively easier to gather than infrastructure and information system data items. Thus, an automation of this collection might be of lower value since it saves less time.

In addition we wanted to find out which types of data sources the participants deemed important. Therefore, we presented them a list of possible information sources to rank them. The results are depicted in Figure 6. It can be seen that the integration of CMDBs has the highest interest among the participants. In addition the integration of business process engines and existing databases rank almost equally high. It is also interesting that almost one quarter of the participants selected the "Don't know" option regarding the integration of Cloud APIs. A reason for this might be that the concept of cloud computing for infrastructure provision has not fully arrived in the mainstream yet and not used in all organizations. We have presented a process for integrating infrastructure cloud instances into enterprise architecture management in (Farwick et al., 2010). In addition, it can be stated that no integration source seems to be unimportant in general.

The participants had the opportunity to add their own suggestions on possible information sources. Among these were Enterprise Service Bus Configurations (interfaces), monitoring tools like Nagios[4], organizational information from governance and organization management tools, financial planning systems, mainframe applications, and Microsoft Visio.

Data Quality Attributes. In order to assess the requirements for data quality of an automated EA maintenance mechanism, we asked the survey participants to rank different data quality attributes. The results are the following:

1. Correct Granularity - data is represented at the desired level of abstraction (Score 75).
2. Consistency - data contains no errors and contradictions (Score 73).
3. Actuality - data is up-to-date (Score 60).
4. Completeness - data covers all desired elements (Score 52).

The results show that the correct granularity and consistency of the data in an EA repository are valued highest by the participants. One can see that correct and consistent data is much more important than very current information. The participants were also able

[4]http://www.nagios.org last retrieved 1/24/11

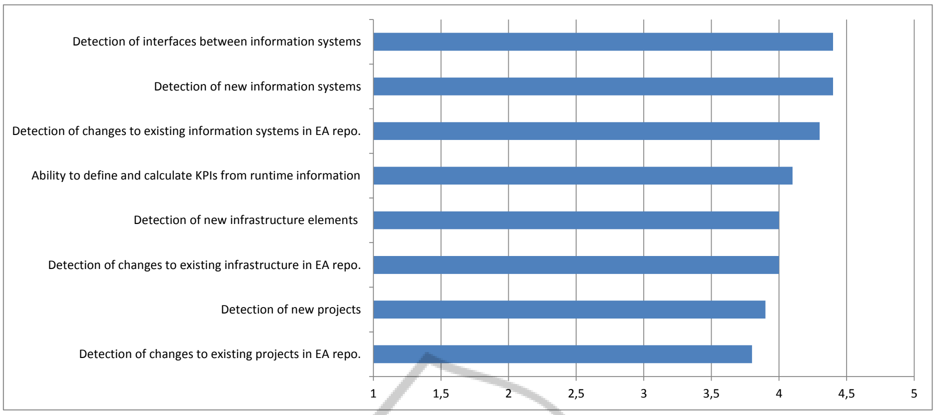

Figure 5: Average subjective usefulness of selected features, where 1 corresponds to "Not useful" and 5 corresponds to "Very useful".

to suggest own data quality attributes that they consider important. This resulted in the suggested data quality attributes of being able to find out the data owner, and the ability to determine the source for a datum.

In this section we have discussed the results of our survey on automated EA maintenance. As a final remark it can be stated that the general response to the idea was positive and that the majority of the respondents are in demand for automation. However, several participants expressed doubts whether the investment in automated EA maintenance will result in sufficient value. We fully agree that it is important to calculate the value of such a solution, therefore we present success evaluation criteria in Section 6, after we described our collection of requirements for automated EA maintenance in the next section.

5 REQUIREMENTS FOR AUTOMATED EA MODEL MAINTENANCE

The general findings of the survey already indicate that automated model maintenance could reduce EA modeling efforts, and that it would contribute to better meet desired EA element coverages. This, and the related work discussed in Section 3, underlines our assumption that automated EA maintenance is a relevant need in practice and that current tool support does not fulfill this need in a satisfactory manner.

As a first step in the direction of an EA tool that aids EA practitioners to collect relevant EA data from the enterprise environment, we present a set of requirements for such a tool in this section. Our basic vision is a system which automatically collects data from various sources into a central repository. These sources could be runtime systems, such as application servers and information systems like project portfolio management tools, or low–level sensors that monitor infrastructure elements. In addition, the system should help in keeping the data quality high and provide a computer aided process to guide manual and automated data collection.

The requirements presented in this section were derived from the following information sources:

- Related work (see Section 3).
- Our own experience as EA consultants.
- Input from users of EA tool iteraplan.
- Results of our survey.

Table 1 summarizes these requirements which we have split into six categories. The table also lists the sources of the specific requirements. In the following we discuss the requirements of each section in detail.

5.1 Architectural Requirements (AR)

The architectural requirement stems from the heterogeneous environment that most enterprises are currently faced with. It is supposed to help integrating the system into enterprises that have physically distributed departments which can hinder the information flow.

AR 1 - The Collection of EA Data must be Federated from the Repositories of the Data Owners. This requirement is inspired by (Fischer et al., 2007). They argue that if data is collected at the side of the data owners, these owners can use the data collection and modeling tools they are familiar with. The automation mechanism must then collect and integrate the data into the central EA repository. Another publication supporting this requirement is (Breu, 2010).

Table 1: Requirements for automated EA models maintenance.

ID	Description	Source
	Architectural Requirements (AR)	
AR 1	The collection of EA data must be federated from the repositories of the data owners (departments etc.)	Survey & (Fischer et al., 2007; Farwick et al., 2010) & own experience
	Organizational Requirements (OR)	
OR 1	An organizational process must be in place that regulates the maintenance of EA models	(Fischer et al., 2007; Moser et al., 2009; Hanschke, 2009) & own experience
OR 1.1	The organizational maintenance process must be supported by a technical process	Implied by OR 1
OR 1.2	The system must be able to adapt the maintenance process to the existing processes in a company	Implied by OR 1.1
OR 2	Each data source must have an owner/responsible	(Fischer et al., 2007; Hanschke, 2009)
OR 2.1	The technical maintenance process must allow for delegation of rights to ensure that the QA process is always executable	Implied by OR 2
	Integration/Data Source Requirements (IR)	
IR 1	The system must be able to detect changes in the real world enterprise architecture	Survey & (Moser et al., 2009; ter Doest and Lankhorst, 2004) & own experience
IR 1.1	The system must provide a mechanism to define the mapping from incoming data to the internal data structure	(Moser et al., 2009) implied by IR 1
IR 1.2	The system must be able to detect changes to the infrastructure	Survey & own experience
IR 1.3	The system must be able to detect interfaces between information systems	Survey & own experience
IR 1.4	The system must be able to detect changes to information systems	Survey & own experience
IR 1.5	The system must be able to integrate information about projects	Survey & own experience & (Buckl et al., 2008)
IR 2	The system must have a machine understandable internal data structure	(Tanner and Feridun, 2009)
IR 2.1	The system must be able to be configured to transform incoming data, to the internal machine understandable data format	Implied by IR 2
	Data Quality Requirements (DQR)	
DQR 1	The system must provide mechanisms that help the QA team to ensure data consistency	Survey & (Hafner and Winter, 2008)
DQR 2	The system must provide mechanisms to ensure data actuality that is sufficient for the EA goals	Survey & own experience & (Hanschke, 2009)
DQR 2.1	Each element in the systems data structure must have a creation time stamp and an expiration date (volatility)	Implied by DQR 2
DQR 3	The system must provide mechanisms to adjust the granularity of data	Survey & Implied by IR 1.1
DQR 4	The system must provide mechanisms that allow for the automated propagation of changes	(Dam et al., 2010)
DQR 5	The system must be able to identify and resolve data identity conflicts from different sources via identity reconciliation	Implied by AR 1 & (Fischer et al., 2007)
	Functional System Requirements (FR)	
FR 1	The system must allow for the definition of KPIs calculations	Survey & (Hafner and Winter, 2008)
FR 2	The system must be able to calculate the defined KPIs from runtime information	Survey & (Hafner and Winter, 2008)
	Non-functional Requirements (NFR)	
NFR 1	The system must scale for large data input	Own experience & (Hafner and Winter, 2008)

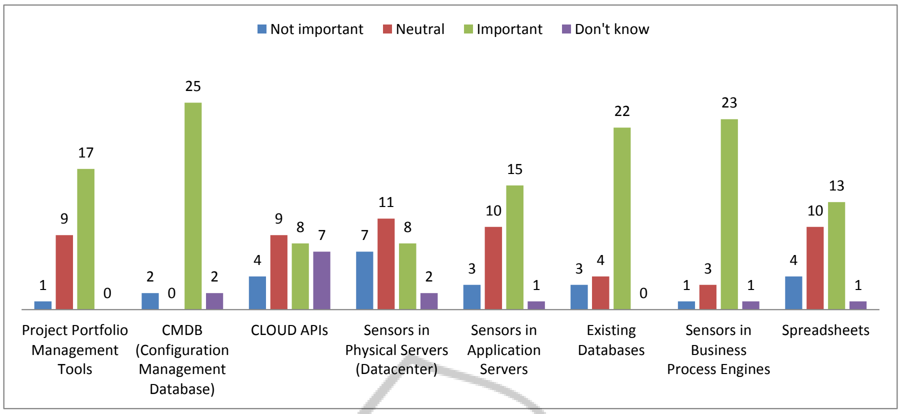

Figure 6: Answers to the question “Which information sources to automatically feed data into an EA repository do you consider important?”.

5.2 Organizational Requirements

The following requirements stem from acknowledging that a full automation without human intervention is unrealistic. In addition, a key aspect of the EA practice is modeling, which is a creative task that cannot be done by machines. Therefore, the following requirements put EA maintenance in its organizational context, by identifying stakeholders and owners as well as defining organizational processes.

OR 1 - An Organizational Process must be in Place that Regulates the Maintenance of EA Models. This basic requirement that has been stated in (Fischer et al., 2007; Moser et al., 2009) requires an organizational process in which the responsibilities of the federated data providers and the data owners are defined.

OR 1.1 - The Organizational Maintenance Process must be Supported by a Technical Process. This requirement is implied by OR 1 since the organizational process needs to be guided by a technical process that helps the process participants to fulfill required tasks and perform maintenance activities in the desired intervals. With technical process we mean that the organizational process is actually executed as a workflow in a business process engine, that presents all stakeholders the tasks they need to execute in a task list.

OR 1.2 - The System must be Able to Adapt the Maintenance Process to the Existing Processes in a Company. The technical process of OR 1.1 must be adaptable in such a way that it can be integrated into the existing organizational processes of an enterprise.

OR 2 - Each Data Source must have an Owner/Responsible. This requirement demands the assignment of ownership to a data source. This enables the recognition of the source of each data item in the repository which is an important aspect of the quality assurance process of the EA data (see data quality requirements Section 5.4). For example, it is needed if new data is pushed from a federated data source to the central EA repository for data integration and data conflicts occur that cannot be resolved automatically. In this case the data owner could be notified by the automated QA process to resolve the conflict.

OR 2.1 - The Technical Maintenance Process must Allow for Delegation of Rights to Ensure that the QA Process is Always Executable. This requirement is implied by the requirement OR 2. In case a specific data owner cannot fulfill the task of resolving a data conflict issue, she needs to be able to delegate the right of modifying the data to another person. Thus, the QA process can proceed.

5.3 Integration/Data Source Requirements

The following requirements relate to the types of data sources and the type of data format that is needed.

IR 1 - The System must be Able to Detect Changes in the Real World Enterprise Architecture. As a basic requirement, the desired system must be able to detect changes in the real enterprise and relate those to the the elements in the EA repository. This is one of the essential features expressed in Doest & Lankhorst’s vision of future EA tools support in (ter Doest and Lankhorst, 2004). The information sources could potentially be anything from low–level infrastructure information, over information from release and license management tools, up to high–level governance tools. The importance here is that the data can potentially be abstracted and combined into a granularity that is appropriate for EAM.

IR 1.1 - The System must Provide a Mechanism to Define the Mapping from Incoming Data to the Internal Data Structure. This requirement is implied by requirement IR 1 since it is unrealistic to assume that all federated data sources expose the same data schema as the central EA repository (Moser et al., 2009).

IR 1.2 - The System must be Able to Detect Changes to the Infrastructure. As it can be seen in the result of the survey presented in the previous section, the discrepancy between the amount of infrastructure elements in the repositories, and the amount that should optimally be covered is the highest among the survey participants. Therefore, it is important that automated detection of infrastructure changes are included in a solution.

IR 1.3 - The System must be Able to Detect Interfaces between Information Systems. As it can be seen in the results from our survey presented in Section 4, the automated detection of interfaces between information systems had the highest value for the survey participants. Therefore, an automation method needs to be provided that can infer the interfaces between information systems on a high level of abstraction.

IR 1.4 - The System must be Able to Detect Changes to Information Systems. Similar to requirement IR 1.3 this requirement seemed to have much value to the survey participants. More specific requirements for this are also defined in (Hafner and Winter, 2008)

IR 1.5 - The System must be Able to Integrate Information about Projects. It is basic practice in EA to relate projects in an organization to architecture elements that are affected by the project. This requirement is also identified in (Buckl et al., 2008).

IR 2 - The System must have a Machine-understandable Internal Data Structure. In order to solve semantic heterogeneity between information sources and to facilitate data quality assurance, the system should have an internal data structure that is machine–understandable. This kind of semantic integration is described in (Tanner and Feridun, 2009).

5.4 Data Quality Requirements

Data quality is a major concern in the practice of EA since the decisions made on the basis of EA data can only be as good as the data they are based on. Therefore, this section describes several data quality–related requirements. Although, the quality requirements can also be seen as non-functional requirements (see Section 5.6), they are listed in their own category because of their important role in EAM.

DQR 1 - The System must Provide Mechanisms that Help the Quality Assurance Responsibles to Ensure Data Consistency. In a system which integrates data from various sources, data consistency is a major concern. Therefore, it needs to be specifically addressed as a requirement. This requirement relates to requirement OR 1.1 in the sense that the QA process should be embedded into the general automated EA data collection process.

DQR 2 - The System must Provide Mechanisms to Ensure Data Actuality that is Sufficient for the EA Goals. In order to keep the quality of the EA data high, the maintenance process must provide a mechanism that checks the actuality of data items. In case the actuality of some items is not good enough, an update process should be initiated.

DQR 2.1 - Each Element in the System's Data Structure must have a Creation Time Stamp and an Expiration Date (Volatility). As a result of requirement DQR 2 each data item needs to have meta–data attached to it that indicates the creation time stamp and a form of expiration date. Different types of data may have a different volatility, e.g. the past has shown that mainframe servers have a longer life span than a specific version of an information system. Thus, each information source and information type must have a specific expiration time span.

DQR 3 - The System must Provide Mechanisms to Adjust the Granularity of the Data in the Repository. The correct granularity of data is crucial to realize the benefit of EA. If the data is too fine-grained, it is hard to draw strategic conclusions from it, if it is too coarse-grained, the information contained in it is too unspecific. In a solution that includes automated data retrieval it is likely that the collected data is finer grained than what is needed by the EA practitioner.

Thus, the system must provide a filter mechanism that can adjust the information that is actually displayed in the views.

DQR 4 - The System must Provide Mechanisms that Allow for the Automated Propagation of Changes. As identified by several research publications, such as (Dam et al., 2010) and (Kumar et al., 2008), it is often the case that changes to the EA repository data imply additional changes, to keep the data model consistent. Hence, the system should provide mechanisms to identify demand for those secondary changes.

DQR 5 - The System must be Able to Identify and Resolve Data Identity Conflicts from Different Sources Via Identity Reconciliation. A reason for inconsistencies can be if several objects are present in an EA repository that describe the same physical object independently from each other. This problem can occur when several data sources are integrated which do not share object identifiers. Thus, the system must provide means to identify and resolve these object identities.

5.5 Functional System Requirements

FR 1 - The System must Allow for the Definition of KPI Calculations. This requirement demands a language with which the calculation algorithm for Key Performance Indicators (KPI) can be defined.

FR 2 - The System must be Able to Calculate the Defined KPIs from Runtime Information. The ability to gather runtime information also brings the positive effect that the actuality and fine granularity of the collected data allows for up-to-date calculation of KPIs. So this requirement demands that the KPIs defined in FR 1 can be calculated from runtime information.

5.6 Non-functional Requirements

NFR 1 - The System must Scale for Large Amounts of Data Input and Large Repositories. Real enterprises can have thousands of elements in their EA repository. In order to ensure responsiveness of queries and the EA tool in general, care must be taken during the development phase to ensure scalability. This requirement is also stated in (Hafner and Winter, 2008).

In this section we have presented a list of requirements for a method, process and tool that supports automated EA maintenance. In the next section we look at success evaluation criteria that can be applied to calculate the effectiveness of such a solution after it has been implemented and applied.

6 SUCCESS EVALUATION CRITERIA

In order to evaluate the usefulness of an automation solution after it has been implemented, it is important to clarify success evaluation criteria. Establishing such criteria early on, also helps during the implementation of a project to not lose the focus of what is actually demanded by all stakeholders. Please note that these success criteria do not directly relate to the requirements from the previous section, but rather refer to the success of an automated EA maintenance effort in general. Overall, it can be said that it is likely that EA efforts which have a stronger focus on the IT side, than on the business side, might see a larger benefit from such a solution, because collecting high quality data of the infrastructure is more difficult than collecting information about the business layers. In the categorization of EA situations in enterprises according to (Saat et al., 2011), these companies fall under the *Technical Quality Biased* and *Aligned Innovation Biased* categories.

In the following we introduce evaluation criteria for an automated EA maintenance mechanism, and group these by stakeholders that are potentially affected by such an implementation. We identified three types of stakeholders that are directly or indirectly involved with an automated EA maintenance method. These are enterprise architects, data providers/owners, and the upper level management, such as the CIO.

Success Criteria for Enterprise Architects. The enterprise architects are the stakeholder group which can benefit the most from the automated EA maintenance. If implemented the right way, the system should allow the EA architects to focus on the core EA tasks, i.e. analyzing the current architecture, planning the to-be architecture and ensuring that the right decisions are made towards this planned architecture. Thus, the key success criteria, from the point of view of enterprise architects, is how much time the system can save them, that would have been used to manually collect and update the EA repository with the former system. Another important aspect is whether more information than before can actually be gathered, i.e. can the system provide a more complete picture of the architecture than the former system. To measure this criterion the enterprise architecture model prior to the introduction of the system needs to be compared to the model of the new system. With this comparison it would be possible to see, if, for example, the coverage of information systems or infrastructure elements has increased. In addition, all data quality attributes (see Section 5.4) need to be evaluated in order measure the effectiveness of the approach.

Success Criteria for Data Providers. A critical part for the success of such a system is whether the data owners and data providers are willing or capable of providing the needed data in a satisfactory quality. That is, the data owners need to actively take part in the provision of data and the quality assurance process that is initiated after data has been inserted into the central EA repository. Hence, one success criterion is which and how many data sources (e.g. departments) provide interfaces to automatically collect the data. It is therefore essential for the success that providing the data by the data owners is facilitated as much as possible to get their support. It needs to be measured to what degree the automation process can actually be integrated into the normal work processes of the data owners.

Another critical aspect of the data integration is that integration work has to be conducted to include various data sources. This can be a laborious task by itself, and the integration implementation also needs to be maintained. Or, as one of our survey participants stated: *"(A) challenge will be to enable 'just enough' automated maintenance and avoid overengineering"*. Thus, a balance needs to be found between the creation of integration mechanisms and manual collection of data, in order not to waste resources on integration. This amount of integration effort needs to be measured and evaluated against the benefits.

Success Criteria for Management. Another group of beneficiaries are members of the upper–level management such as the CIO. From the perspective of this stakeholder group the automated EA maintenance effort is a success if the provided data quality is raised. By this, this group can make strategic decisions based on more accurate data. In order to measure the success from the management point of view the data quality attributes need to be measured similar to the evaluation for the enterprise architects.

To summarize, the success evaluation criteria from the above paragraphs, it can be said that the overall goal is achieved, if the manual collection of EA data is significantly reduced and the data quality is raised, while the additional work for the data providers and systems operation teams can be kept at a minimum.

7 CONCLUSIONS & FUTURE WORK

Enterprise architecture management is the practice of modeling the business and IT components of an enterprise and relating them to each other. Thereby, the dependencies between the business and the underlying IT support can be analyzed, and strategic decisions can be made towards a consolidated architecture that matches the business needs. Enterprise architecture models can become very large, and thus the creation and maintenance of the EA models is often time consuming and costly. Therefore, the questions arise if there exists literature that discusses (automated) EA maintenance, to what degree companies actually use automation techniques and what requirements for an automated EA mechanism there are.

In this paper, we first looked at related work in the field of EA in general and EA maintenance processes. We discovered that there does not exist related work that directly addresses the problem of automated EA maintenance. Then, we presented the results of a survey we conducted among EA practitioners to identify how EA maintenance is actually executed in companies today, and what requirements they have in terms of *automated* EA maintenance. As the result of the survey, the analysis of related work, and own experiences, we then presented a list of requirements for a tool and a method to facilitate automated EA maintenance. Finally, we discussed possible success evaluation criteria for such a solution.

We see the requirements analysis in this paper as the starting point for further development of a method and technical solution for automated EA maintenance. The next step will be to propose an overall technical architecture and to focus on technical issues such as object identity reconciliation, semantic integration and a technical process to support the integration process. In addition, we will extend existing EA maintenance processes such as the processes defined in (Fischer et al., 2007), (Hafner and Winter, 2008) and (Moser et al., 2009) with our ideas on automated EA. Our long-term goal is to integrate the solution into the open-source EA tool iteraplan.

REFERENCES

Ambler, S. W. (2010). Enterprise architecture survey results. http://www.ambysoft.com/surveys/stateOfITUn ion201001.html last retrieved 12/29/2010.

Breu, R. (2010). Ten principles for living models - a manifesto of change-driven software engineering. In *4th International Conference on Complex, Intelligent and Software Intensive Systems (CISIS-2010)*.

Buckl, S., Dierl, T., Matthes, F., Ramacher, R., and Schweda, C. (2008). Current and future tool support for ea management. In *Proceedings of Workshop MDD, SOA und IT-Management (MSI 2008), Oldenburg, Gito*.

Buckl, S., Ernst, A., Matthes, F., and Schweda, C. (2009a). An information model for managed application landscape evolution. *Journal of Enterprise Architecture*, 5(1):12–26.

Buckl, S., Ernst, A. M., Lankes, J., Matthes, F., and Schweda, C. M. (2009b). State of the art in enterprise architecture management. Technical report, Chair for Informatics 19, Technical University Munich, Germany.

Buckl, S., Matthes, F., and Schweda, C. M. (2010). *Future Research Topics in Enterprise Architecture Management–A Knowledge Management Perspective*, pages 1–11. Springer.

Chief Information Officers Council (2007). FEA Consolidated Reference Model Document Version 2.3.

Dam, H. K., Le, L.-S., and Ghose, A. (2010). Supporting change propagation in the evolution of enterprise architectures. In *2010 14th IEEE International Enterprise Distributed Object Computing Conference*, pages 24–33. IEEE.

De Boer, F.S., Bonsangue, M., Groenewegen, L., Stam, A., Stevens, S., and Van Der Torre, L. (2005). Change impact analysis of enterprise architectures. In *Information Reuse and Integration, Conf, 2005. IRI-2005 IEEE International Conference on.*, pages 177–181. IEEE.

DMTF (2010). Configuration management database (cmdb) federation specification - dsp0252 1.0.1.

Farwick, M., Agreiter, B., Breu, R., Ha andring, M., Voges, K., and Hanschke, I. (2010). Towards living landscape models: Automated integration of infrastructure cloud in enterprise architecture management. In *Cloud Computing (CLOUD), 2010 IEEE 3rd International Conference on*, pages 35–42.

Fischer, R., Aier, S., and Winter, R. (2007). A federated approach to enterprise architecture model maintenance. *Enterprise Modelling and Information Systems Architectures*, 2(2):14–22.

Hafner, M. and Winter, R. (2008). Processes for enterprise application architecture management. In *Hawaii International Conference on System Sciences, Proceedings of the 41st Annual*. IEEE.

Hanschke, I. (2009). *Strategic IT Management: A Toolkit for Enterprise Architecture Management*. Springer.

IEEE Computer Society (2000). Ieee recommended practice for architectural description of software intensive systems (ieee std 1471-2000).

Konstantinou, A. and Yemini, Y. (2009). A2A: An architecture for autonomic management coordination. *Integrated Management of Systems, Services, Processes and People in IT*, pages 85–98.

Kumar, A., Raghavan, P., Ramanathan, J., and Ramnath, R. (2008). Enterprise interaction ontology for change impact analysis of complex systems. In *2008 IEEE Asia-Pacific Services Computing Conference*, pages 303–309. Ieee.

Lankhorst, M. (2005). *Enterprise Architecture at Work: Modelling, Communication and Analysis*. Springer, Berlin.

Moser, C., Junginger, S., Brückmann, M., , and Schöne, K. (2009). Some process patterns for enterprise architecture management. In *Proceedings, Workshop on Patterns in Enterprise Architecture Management (PEAM2009), Bonn*, pages 19–30.

OGC (2007). *ITIL Lifecycle Publication Suite Books, 2nd impression*. TSO.

Saat, J., Winter, R., Franke, U., Lagerstroem, R., and Ekstedt, M. (2011). Analysis of it/business alignment situations as a precondition for the design and engineering of situated it/business alignment solutions. In *Proc. Hawaii International Conference on System Sciences (HICSS-44).(January 2011)*.

Tanner, A. and Feridun, M. (2009). Fusio: Semantic integration of systems management and enterprise information. Technical report, IBM Research GmbH Zurich Research Laboratory.

ter Doest, H. and Lankhorst, M. (2004). Tool support for enterprise architecture-a vision. Technical report, Telematica Instituut, Enschede.

The Open Group (2009). The Open Group Architecture Framework (TOGAF) version 9.

US Department of Defense (2010). The DoDAF Architecture Framework Version 2.02.

Wilson, C. (2010). Gartner - magic quadrant for enterprise architecture tools.

Winter, K., Buckl, S., Matthes, F., and Schweda, C. (2010). Investigating the state-of-the-art in enterprise architecture management methods in literature and practice. In *MCIS 2010 Proceedings*.

Winter, R. and Fischer, R. (2007). Essential layers, artifacts, and dependencies of enterprise architecture. *Journal of Enterprise Architecture*, pages 1–12.

Zachman, J. A. (1987). A framework for information systems architecture. *IBM Systems Journal*, 26(3):276–292.