Processes for Enterprise Application Architecture Management

Martin Hafner
Institute of Information Management,
University of St. Gallen, Switzerland
Martin.Hafner@unisg.ch

Robert Winter
Institute of Information Management,
University of St. Gallen, Switzerland
Robert.Winter@unisg.ch

Abstract

Over the past few decades, the way in which enterprises have performed their business and organizational changes at the level of information systems has for the most part been incomplete and / or unsystematic. Fundamental IT innovations have therefore coincided with the development of heterogeneous application landscapes which are more or less inconsistent with the business and / or process architecture. Explicit management of the application architecture, which forms the interface between the business and the technical view on the information system, is necessary to recreate and preserve consistency. Based on a requirements analysis for enterprise application architecture management and a discussion of related work from literature and practice, this paper proposes processes that are based on three case studies. The proposed processes are evaluated in respect of the specified requirements.

1. Introduction

1.1. Enterprise Modeling and IT Business Alignment

Complex systems can be modeled from a wide range of different views and for a wide range of different purposes. In the case of enterprise modeling, forming a hierarchy of architecture *layers*, which is often supplemented by the separation of different architecture views, has proved successful in mastering complexity. Architecture layers are distinguished by their level of aggregation, their level of generalization, their implementation-independence and / or their respective design goals. Models at different architecture layers and models at one architecture layer, which have different degrees of aggregation or generalization, can be hierarchically linked in order to ensure consistency in the overall depiction. In contrast, architecture *views* do not imply any hierarchization but instead represent partial models which concentrate on a specific modeling aspect in order to reduce model complexity.

Although widely used enterprise modeling approaches like "The Open Group Architecture Framework" [15], "Federal Enterprise Architecture Framework" [4], or the "Zachman framework" [26] have significant differences regarding comprised architecture layers and artifact types, a common understanding of enterprise architectures becomes nevertheless evident: [2]

- **Business architecture** represents an aggregated, enterprise wide model of service exchanges and financial flows in value networks, strategic positioning of business units, product / service specifications, organizational goals and success factors as well as dependencies between these artifacts.
- **Process architecture** represents an aggregate, enterprise wide model of service development, service creation and service distribution activities, of organizational units, of information requirements and of operational process management as well as dependencies between these artifacts.
- **Application architecture** represent an aggregate, enterprise wide model of logical functionality clusters (applications) as well as information flows and control flows between applications.
- **Software architecture** represents an aggregate, enterprise wide model of software artifacts and data structures as well as data and control flows between these artifacts.
- **IT architecture** represents an aggregate, enterprise wide model of hardware and communications components as well as dependencies between these artifacts.

For alignment purposes, it is important to represent not only intra-layer dependencies between architecture artifacts, but also dependencies between artifacts on different architecture layers. According to the hierarchical, multi-level systems theory, design / evolution results on each architecture layer reduce the degrees of freedom of the subsequent layers [13]. Common to all enterprise architecture approaches is the fact that the design of IT related artifacts follows business requirements.

Following innovation phases – such as for example the widespread deployment of first generation information systems in the 1960s and 1970s – the differing life cycle lengths of business and process architectures on the one hand, and IT as well as software architectures on the other causes the structures implemented to slowly but surely gape apart. At the present point in time, networking

and specialization strategies as well as the respective reorganization programs on the one hand reflect state-of-the-art business concepts to be in place in many companies. On the other hand, there are complaints of an IT infrastructure which is perceived as being outmoded and which is becoming increasingly more complex and heterogeneous as a result of permanent redesigns. While increasing complexity and heterogeneity drives IT operations costs upwards, outmoded architectures prevent the consistent implementation of modern business requirements such as for example market coordination between business units, multi-sourcing, real-time management or sometimes even process orientation [24].

1.2. Objectives and Overview

Generally speaking, the goal of application architecture management is the effective and efficient coordination of business and process architectures on the one hand, and software and IT architecture on the other. As a result of the many possible modeling layers, modeling views and design goals, a wide variety of approaches to application architecture management exist both in the literature and in business practice. Since many publications focus on design / evolution principles (like e.g., service orientation) and / or on modeling techniques (e.g., application architecture models), design / evolution goals as well as design / evolution processes are mostly not fully explicated.

The aim of this article is to develop a consolidated process reference model for the managed evolution of application architecture. The focus is limited to the application layer because, as the interface between the business oriented architecture layers and the IT oriented architecture layers, it is of particular importance for an effective and efficient IT business alignment.

The first step is to analyze how management of the application architecture is to be positioned within information management in order to be able to derive concrete objectives and framework conditions. On this basis it will then be possible to select concrete approaches to application architecture management. A critical discussion and comparison of the selected approaches from literature and practice form the basis for the derivation of a generic model of architecture processes.

Following this introduction, requirements to be met by any concept for application architecture management are derived in section 2. In the light of these criteria, the state of the art of application architecture management is discussed in section 0. In order to supplement and extend the state of the art, particularly with regard to the practicability of implementation and communication, three approaches to application architecture management taken from the world of practice are analyzed based on their architecture concept and architecture management processes in section 4. The comparison of these practical approaches forms the basis for their consolidation into the proposal of a reference process model in section 5. Finally, an evaluation of the proposed reference processes and an outlook for future research are provided in section 6.

2. Goals of Application Architecture Management

The application architecture serves as a transparent communication and design / evolution platform between the various IT stakeholders (e.g., application development sponsors in business and application developers in IT).

The most general foundation for deriving goals of application architecture management is the system of *general organizational objectives*, for example sustainability, speed / agility, quality, and operational efficiency [16]. Since information management (and thus information systems management as well) constitutes a support function in the enterprise, these general goals are specified by the *material and formal goals of information management*, for example information systems maintainability (concretizes sustainability), costs and resource optimization (concretize operational efficiency), operating performance and agility (concretize speed) as well as security, transparency and reliability (concretize quality). As a means for information management, *architecture management goals* can be further concretized on that basis. Enterprise architecture management should be aimed at [1, 11]:

- integrating business requirements and IT potentials / restrictions,
- preserving consistency of designs on the business-related layers and the IT-related layers of enterprise architecture,
- being situation oriented (rather than trying to deploy standardized solutions),
- creating and maintaining transparency, and
- supporting agility (i.e., supporting designs that are easy to adapt).

Based on [1], these general goals of enterprise architecture management have been transformed into requirements for application architecture management. In Table 1, these requirements are complemented by some general requirements.

Due to their particular importance for the general positioning of application architecture management, R1, R2 and R4 are used as mandatory requirements for the discussion of the state-of-the-art (i.e. these determine whether an approach is regarded at all), while the other requirements are used for evaluation of the approaches that are compatible with R1, R2 and R4.

Table 1. Requirements for Application Architecture Management Approaches

Designation	*Description*	*Source*	*Type*
R1	*Since application architecture takes on a crucial role for IT business alignment, an approach must* ***explicitly address application architecture management****.*	*Section 2*	*Mandatory*
R2	*In order to operationalize architecture management and to embed it in the organization, an approach must propose a* ***detailed process model****.*	*[3]*	*Mandatory*
R3	*Architecture management should be* ***scalable*** *with increasing requirements.*	*Scalability [1]*	*Evaluation*
R4	*Architecture management must have an* ***evolutionary character*** *as its influencing factors mean that it must continually balance the long-term alignment of the architecture against short-term entrepreneurial success.*	*Evolutionary capability [1, 11]*	*Mandatory*
R5	*Architecture management should be* ***organizationally compatible*** *with its associated tasks, particularly in information systems management.*	*Comprehensibility [1]*	*Evaluation*
R6	*Architecture management should be able to corroborate its effectiveness and efficiency in the form of* ***performance indicators****.*	*Operational efficiency [1]*	*Evaluation*
R7	*Architecture management should produce* ***methodological results*** *in the form of architecture artifacts such as models, standards, etc.*	*Comprehensibility [1, 11]*	*Evaluation*
R8	*Architecture management should allow for the constant analysis of its* ***influencing factors*** *and its long-term objectives.*	*Complexity [1]*	*Evaluation*
R9	*Architecture management should allow for the development of* ***visions for to-be architecture alternative****s.*	*Stability and encapsulation [1]*	*Evaluation*
R10	*Obviously, it is not possible to assume a consistent enterprise architecture at a specific point in time, which means there should be a prime focus on* ***dealing with inconsistencies*** *[14, 18].*	*Stability [1]*	*Evaluation*
R11	*While maintaining an exchange with its stakeholders and associated task areas, application architecture management should adopt a* ***service-oriented approach*** *in the virtual absence of other options for pushing through its point of view or of other benefit arguments.*	*Consensus and customer orientation [1]*	*Evaluation*

In the following sections, the stated requirements are applied to existing process models for application architecture management (section 0), case studies (section 4) and the proposal for a corresponding consolidated reference model (sections 5 and 6).

3. Existing Process Models for Application Architecture Management

This section gives an overview of different approaches to application architecture management, their respective process models and their suitability in respect of the requirements specified in section 2. Only approaches which meet the mandatory criteria from Table 1 are considered. The evolutionary character must at least be assured by a continuous management cycle which, in line with common standardization processes [10], allows to include requirements for the evolution of architecture artifacts. Consequently, approaches that mainly focus on enterprise architecture modeling or concentrate on static architecture aspects (i.e., enterprise or information system structures) [8] are not considered here. When describing the approaches, the original terms employed by the respective authors are used. The frequent use of "IS architecture" or even "IT architecture" points to the fact that application architecture is frequently only marginally included in the thinking behind the approaches.

IBM's "Enterprise Architecture Management" process model [23] focuses on the enterprise architecture as a whole and envisages five partial processes for architecture management: (a) update of the architecture in the light of business requirements, (b) architecture review to ensure that architecture-related subject areas comply with strategy, (c) identification of development in the area of business and IT strategy, (d) targeted approval of individ-

ual inconsistencies, and (e) communication of the importance of architectures. The identification of developments is part of a bidirectional exchange with all partial processes; the updating of the architecture affects the architecture review. The process model is not publicly available in detail, which means that little can be said about its relevance to practice. The question of scalability also remains unanswered. Furthermore, proof of the effectiveness and efficiency of the approach, the inclusion of operational business and IT-specific requirements as well as explicit service orientation are not reflected in the model.

Perks and Beveridge [19] use the Architectural Development Method (ADM) for the process model of their approach "Guide to Enterprise IT Architecture". This is performed cyclically in seven partial processes and focuses on the development of an organization's technical architecture. While the first six phases cover the linear realization of a target architecture defined at the beginning of the architecture cycle, which should be repeated every three to five years, the seventh phase is dedicated to maintenance of the IT architecture. Here, according to Perks and Beveridge an effort is made to combat the constant "erosion" of the architecture by observing developments and consistently addressing what is referred to as "architectural drift". This means that minor further developments or arbitrary changes are always performed within the framework of an architecture-driven IT governance process or else prevented. A major drift initiates an ADM cycle restart where the entire architecture is once again up for discussion. As triggers for changes Perks and Beveridge state business strategy, technology, "chaos", external boundary conditions as well as targeted transformations. Although business influences are taken into account, the approach is primarily geared to technical aspects. As a result of the stringent implementation of architecture goals, the process addresses evolutionary aspects only in the concluding partial process. Even in here, however, the main idea is to eliminate inconsistencies. For this reason, service orientation is barely identifiable with this approach.

According to Dern [5], "IT Architecture Management" is geared to software development and differentiates the phases of architecture planning and architecture development. While the prime emphasis is on analyzing the existing information systems, recurrent requirements to be met by further development are recorded, analyzed and evaluated. Management then decides which requirements will be implemented in the information systems portfolio. Within this framework of action, architecture development focuses on the updating of the IT architecture which is integrated into the software development process. The integrated relationship between software development and IT architecture is dominated by development so the effective implementation of visionary architecture artifacts in particular is questionable.

The "IT Architecture Engineering" approach of Krüger and Seelmann-Eggebert [12] focuses on the interaction with project management. It takes external (market) and internal needs (delta of to-be and as-is architecture) into consideration. Adapted processes or targeted migration efforts in the direction of a to-be architecture give rise to requirements which are included in the project (requirements management). Projects are supported by architecture representatives and evaluated in respect of their conformity. During the course of the project, the application and data architecture (actual architecture management) and subsequently the process and organization structures (organization management) of the affected organization unit are adapted in line with project requirements. Irrespective of this management cycle, which is accompanied by continuous "change controlling", artifacts for the as-is architecture (guidelines, standards, processes, frameworks) are provided within the framework of architecture development, and the to-be architecture including a roadmap elaborated. Moreover, further development of the IT architecture and the organization is not clearly performed on the basis of the totality of all requirements but on individual project results. The process model is not very detailed in respect of actual architecture management.

The emphasis of existing reference process proposals for application architecture management (R1, R2) varies widely. In most cases, these models are not very detailed and neither embedded in the theory nor transparently derived from practice. The evolutionary character (R4) is usually covered by a cyclical process, connectivity (R5) is fulfilled through the analysis of influencing factors (R8) but few partial processes for communicating and enforcing architecture on the basis of service orientation (R11) are directly integrated into the processes. In the majority of cases, inconsistencies (R10) are only avoided through the dominance of development projects over IT or the stringent implementation of long-term architecture goals (R9). The approaches discussed here only address the requirements to be met by application architecture management in parts. As a consequence, we analyze several case studies in order to consolidate real-world application architecture processes with findings from the literature into a reference process model that meets all requirements stated in Table 1.

4. Application Architecture Management Cases

The process model to be developed for application architecture management is transparently derived from the world of practice. Three application architecture management cases are therefore outlined below which were

obtained by means of interviews and document analysis and selected according to the mandatory criteria in Table 1. These examples turned out as good practices because of their relatively high coincidence with R3 and R5 through R11. Their respective underlying concept of architecture (in accordance with R1) and process model itself (R2) are looked at in greater detail. In addition, the phases of architecture management are presented as far as possible in accordance with the companies' own descriptions. The underlying processes which deliver concrete results [7] are then described and suitably laid out. Since the interaction with its target groups is central to evolutionary architecture management (R4), its own support processes are not considered here. Furthermore, it is transparently outlined for the individual real-world processes which of them reflect the evaluation requirements (cf. Table 1) in the authors' view.

4.1. Case A: Credit Suisse

Originating from the Schweizerische Kreditanstalt and having grown in several mergers, Credit Suisse (CS) is now the second largest bank in Switzerland and one of the largest financial service institutions in Europe. At CS, architecture management is divided into the *areas* "application architecture" and "infrastructure architecture" as well as into the cross-sectional areas "integration architecture", "security architecture" and "system management architecture".

The structure of architecture management distinguishes three *phases* across all areas: architecture development, architecture communication and architecture enforcement. The phases are cyclically related (Figure 1) and secure the overall goals of the area IT architecture and standards: strategic flexibility, high IT efficiency and managed evolution.

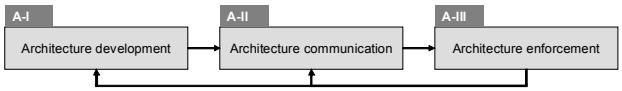

Figure 1. Architecture Management Phases at CS (as presented by CS)

Within the framework of *architecture development*, the current architecture is assessed, i.e. a comparison between the as-is and the to-be architecture is performed for the individual areas mentioned above (Figure 2). Together with relevant developments from the architecture environment (strategy, business departments, technology, IT tasks), which in some cases are proactively identified by architecture management, requirements are derived which are consolidated, assessed and prioritized in respect of any piloting or implementation as architecture artifacts (technology appraisal, principles, standards, roadmaps, standard application platforms, etc.). Depending on the available architecture artifacts, target group-oriented *communication* of the architecture is performed (presentations, training, documents, intranet), the diffusion of which is measured (Figure 2). In the phase of *architecture implementation*, architecture services (consultancy, pre-developed standard application platforms) are provided for projects. Moreover, the project results are checked for architecture conformity and reusability, giving rise to further communication and enforcement activities as well as information for assessing the architecture benefits for projects (Figure 2).

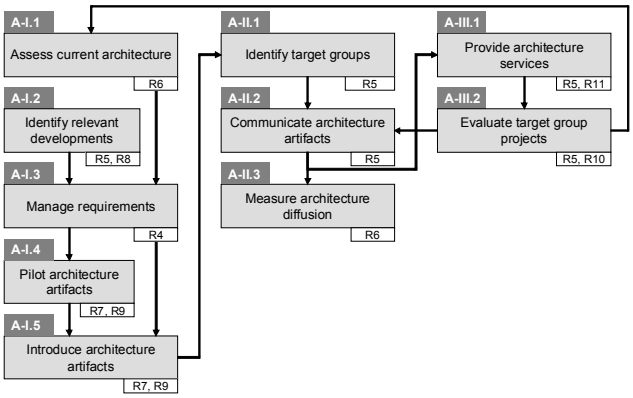

Figure 2. Architecture Management Processes at CS

Figure 2 shows as well which evaluation requirements are related to which activity. Furthermore, R4 (evolutionary character) is fulfilled by the overall feedback loop in the process model. It becomes clear that all minimum requirements have been taken into account and each activity can be traced back to a requirement. An aspect which is to be criticized is the inadequate separation between planning and development processes as architecture assessment and architecture development lie in one phase. Moreover, the extensive organizational effort along the process model can stand in the way of flexibility and scalability of the approach.

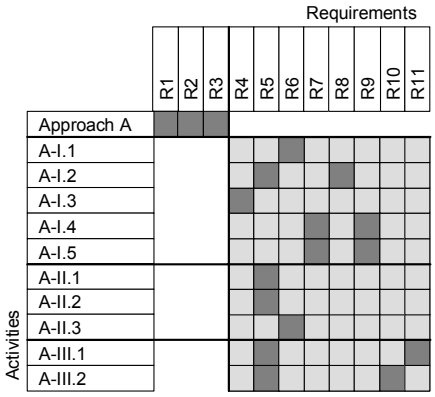

Figure 3. Summary of the Credit Suisse Case

Figure 3 is an overview how the activities of the CS process model satisfy the requirements specified in section 2. The CS process model is very comprehensive with regard to the diversity of processes in architecture management and in this respect provides the framework for the development of a process model for architecture management.

4.2. Case B: Die Mobiliar

'Die Mobiliar' is a large non-life insurance company in Switzerland. Its architecture management [6] differentiates three *architecture layers*: business architecture (business processes, strategies, principles, events), application architecture (correlations of the application landscape to support business processes) and the technical architecture.

As far as the process structure of architecture management is concerned, there are four *phases* B-I through B-IV and two support processes B-V through B-VI (Figure 4). At all three layers of the architecture, the goals transparency, standardization, avoidance of redundancy, support for business planning as well as project consistency and conformity are pursued.

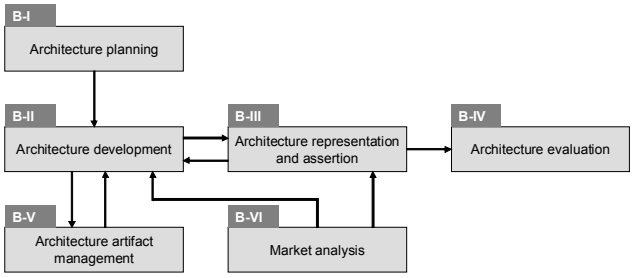

Figure 4. Architecture Management Phases at 'Die Mobiliar'

As a preliminary phase of architecture management, *architecture planning* consists of applying strategic requirements to the existing target architectures (e.g., in the case of the application architecture, by assessing the efficiency and value creation potential of individual applications) in order to ensure their compliance with strategy in the form of architecture principles (Figure 5). As part of *architecture development*, further requirements are considered, in particular those arising from development projects, which are also reflected in the further development of architecture artifacts (business blue-prints, as-is/to-be components of the application architecture, strategies in respect of enterprise architecture, technological partial strategies) as well as appropriate implementation methods (Figure 5). The artifacts are communicated as part of the *architecture representation and assertion*, which is performed primarily during the course of support for projects of the architecture target groups. The operational collaboration of architecture management in projects (architecture office, qualification of project collaborators) leads to the identification of new requirements to be met by architecture management as well as to examination of the conformity of project results measured in terms of the above mentioned objectives of the architecture as part of the *architecture evaluation (*Figure 5*)*. Any conflicts of goals are eliminated as part of a rolling strategic planning.

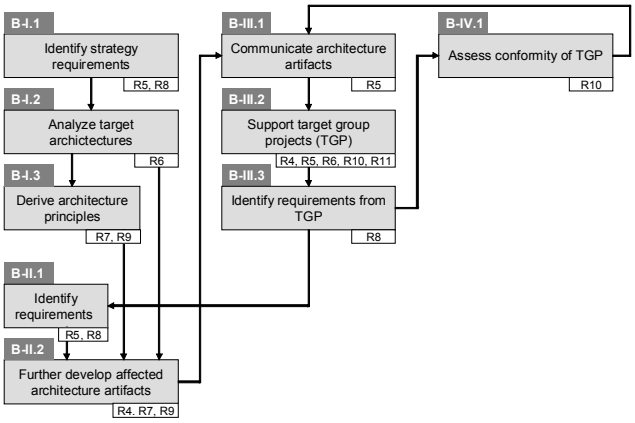

Figure 5. Architecture Management Processes at 'Die Mobiliar'

The striking feature of the 'Die Mobiliar' approach is the pronounced service character of architecture management. Project support is the prerequisite for evolutionary further development, connectivity, enforcement capability and for dealing with inconsistencies between architecture and projects. Above and beyond this, the approach includes the further development of architecture-specific methods in the further development of architecture artifacts, which installs a continuous process of self-renewal.

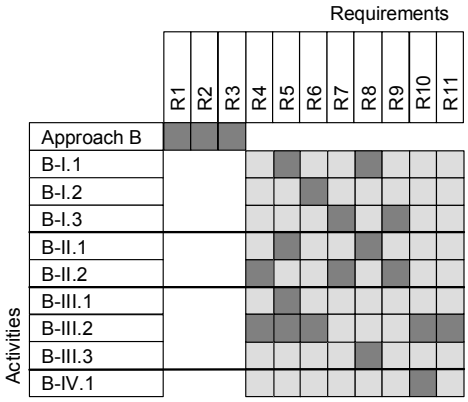

Figure 6. Summary of the 'Die Mobiliar' Case

To summarize, the approach meets all requirements (Figure 6). Here again, the evolutionary character is im-

plemented by management cycles which can be initiated individually depending on concrete needs as and when they arise. The individuality of the approach is an indication of its scalability through the use of additional architects.

4.3. Case C: HypoVereinsbank

As a result of numerous mergers and acquisitions, HypoVereinsbank (HVB) was one of the largest public-owned German banks with a strong presence in Austria and Central Europe at the time of the analysis of its architecture management approach. Recently, HVB was acquired by Italy's Unicredito, a development that is not reflected here.

The *technical architecture* of HVB consists of four layers (application, integration, system, operation layer). The structuring of the application and integration layers is strongly oriented toward the requirements of the business architecture with which they are linked in building blocks.

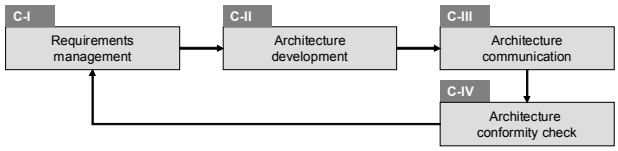

Figure 7. Architecture Management Phases at HVB

Architecture management is implemented by two processes of the same name: one for managing strategic and one for managing operational requirements to be met by the architecture. In addition, architecture management dedicates itself to communicating architecture content via the company's intranet as well as evaluating projects in respect of their architecture conformity. Four *phases* can be identified in total (Figure 7). The goals pursued in this context are in particular transparency, decoupling and standardization in order to satisfy business goals such as cost reduction, flexibility and short time-to-market.

The *requirements* arise on the one hand from the IT strategy, which in the case of HVB is integrated into architecture management, and on the other hand from operational needs such as, for example, those of product managers (Figure 8). The resulting need for action is assessed with regard to architecture conformity, feasibility, impacts and economic benefit and prioritized. In the phase of *architecture development* existing architecture artifacts such as technology sets (product combinations valid for specific use scenarios) are adapted to new requirements and / or extended to include additional artifacts (Figure 8). The *communication* of architecture artifacts is performed primarily via the IT view of an information portal, which in addition to the IT architecture also depicts aspects of process and organizational structure as well as providing documentation standards (Figure 8). In another step, the *conformity* of projects with the specifications of the architecture is checked (Figure 8). The hierarchical level of the body responsible in this case depends on the scope of the project. The frequency of inconsistencies which, if need be, are escalated is thus limited to the cross-organizational rules which are really necessary.

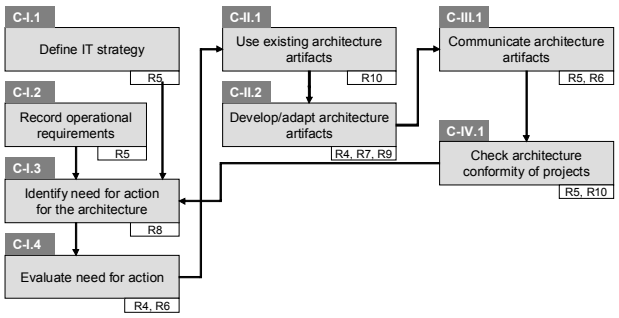

Figure 8. Architecture Management Processes at HVB

In addition to combining architecture specifications and guidelines, the HVB approach is based on a central repository of technology sets. This is used to cover existing needs and / or is systematically extended, thus enabling economies of scale. In addition, the latest models are provided at the layers mentioned above as part of quarterly releases.

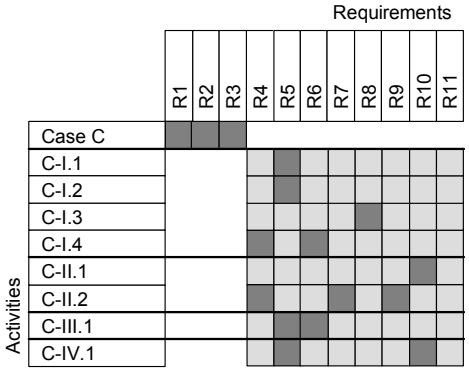

Figure 9. Summary of the HypoVereinsbank Case

In summary, requirements management constitutes the dominant area of architecture management at HVB. As strategic and operational requirements are constantly considered on an integral basis, targeted, evolutionary development is well-established. This is supplemented by company-wide communication and enforcement of architecture specifications. The approach meets the minimum requirements of section 2 although it should address service orientation (R11) more clearly in the cooperation between architecture management and its environment (Figure 9).

5. Consolidated Process Model

5.1. Derivation of the Model

In this section, the case-specific process models presented in section 4 are consolidated into one single process model for application architecture management. The resulting processes can be regarded as reference processes because they are derived from successful practices and can be used as a foundation for company-specific adaptations. Nevertheless, according to [25] it is necessary to define exactly in which degree reference processes can be regarded as universally valid or just as a recommendation. Thus, it has to be clarified in which degree reference models can be directly adopted by an enterprise [22]. In the area of reference modeling mainly domain-specific and domain-independent adoption are distinguished [22]. From the constructivism perspective universal validity can hardly be achieved, because according to constructivists the objective decision whether the reference model can be adopted or not cannot be definitely taken [22] because this depends on individual perception. Nevertheless, the process model to be developed here lays claim to be universally valid over the focused domain. In particular, it is expected to be adoptable for big companies that are characterized by heterogeneously grown application landscapes. Identifying analogous structures and patterns by means of induction is an essential starting point for being accepted as universally valid [21]. As structure and behavior are distinguished in general, structural and behavioral identities must be differentiated [21]. For developing process models mainly behavioral structures are important, whereas for the development of reference models semantic structural analogies play the most important role [21].

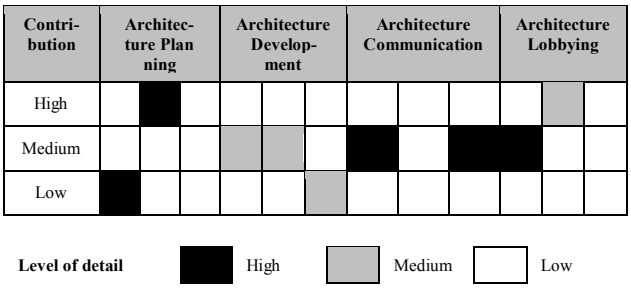

Figure 10. Contribution and Level of Detail of Cases A, B and C for the Consolidated Process Model

Thus, identical structures have to be identified from a semantic point of view. Based on these theoretical considerations, Figure 10 depicts the contribution of the architecture management approaches introduced in section 4 as well as their level of detail.

Figure 11 illustrates the process model that is consolidated from all three approaches. For transparency reasons, references to the respective source processes can be found in the footer of the process descriptions. Since all source processes (from the three approaches) produce results, it is possible to establish semantic comparability and to assume a certain completeness and relevance [20] of the process portfolio. The phases also ensue during the course of consolidation and can be traced back to the finding that further developed architecture artifacts have to be externally communicated and their service orientation put forward to justify their added value. This largely corresponds to the phases of common standardization processes [9]. Above and beyond this, processes are required for continuously monitoring their effectiveness and efficiency [17], for which architecture planning is responsible.

5.2. Discussion of the Phases

In the *architecture planning* phase, strategic requirements are explicitly addressed and existing as-is / to-be and target architectures are assessed for any adaptation requirements. This results in architecture principles which are used during the course of developing further architecture artifacts (frameworks, methods, models, standards, patterns, etc.). In the *architecture development* phase, not only strategic but also operational requirements from IT as well as from the entire enterprise are continuously recorded, consolidated and prioritized, which means that architecture artifacts can be piloted, developed and integrated into the entirety of architecture artifacts. The *architecture communication* phase identifies target groups for training, information material, intranet content, etc. and supplies them with information on the architecture in accordance with their needs. *Architecture lobbying* provides targeted assistance for projects which can be both of a strategic and an operational nature and touch on questions which are relevant to the architecture. In an individual case, this will cover consultancy or direct project collaboration. Furthermore, standardized tool and method components are provided, for example, to relieve development projects of infrastructure details. The ultimate enforcement or acceptance of unavoidable inconsistencies takes place in the assessment of projects. Information from communication of the architecture and its concrete enforcement provides points of reference for assessing diffusion and effectiveness of the architecture. New strategic and operational requirements to be met by the architecture management can arise as a result.

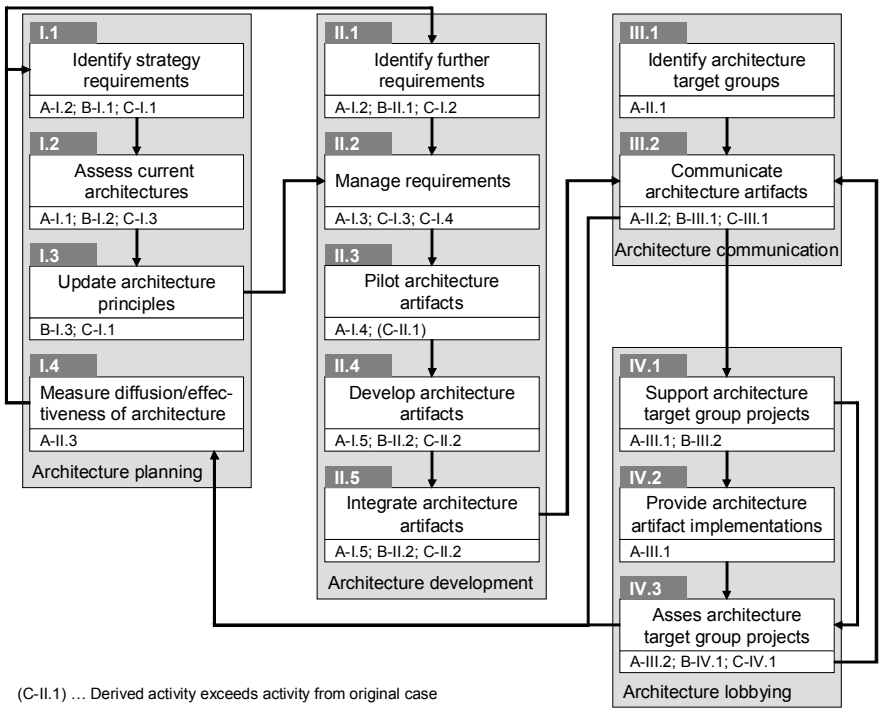

Figure 11. Consolidated Process Model for Architecture Management

6. Conclusion and Outlook

The aim of this article is to identify a process model for application architecture management that can be used as a reference for establishing company-specific architecture management. The basis for that development encompasses the understanding of enterprise and application architecture as documented in section 1 as well as the requirements summarized in section 2. From the large number of proposals for application architecture management, a selection has been discussed that meets mandatory requirements R1, R2 and R4. The discussion shows that the degree to which the assessment requirements can be satisfied is far from complete (section 0). For this reason, three cases from large companies are analyzed and compared in section 4. A consolidated process model is derived from these in section 5. The proposed reference processes address application architecture management (R1), comprise important components of a methodology (R2) and support an incremental architecture evolution (R4). R3 as well as R5 through R11 are secured in particular by management cycles and the differentiated and scalable analysis of requirements (connectivity - R5, analysis of influencing factors - R8), architecture artifacts (methodological results - R7, visions - R9), architecture representation (inconsistency management - R10, connectivity - R5, service orientation - R11) and architecture management (performance indicators - R6). Thus, on the one hand the proposed consolidated process model fulfills the goals of the investigation. On the other hand, the quality of the process model recommendation [22] cannot be definitely verified. According to constructivism, this can only be decided according to the adequacy perceived when the model is adopted in the context of individual circumstances. Schütte emphasizes that reference models are not allowed to be completely built based on a defined system of goals and requirements [21] because goals and in particular their interrelationships highly vary depending on the adoption context of the reference model. This is why the recommendation of the present reference process model cannot be treated as universally valid only regarding the fulfillment of the set of goals and requirements of section 2. For this reason, only the universal validity of the requirements system can be decided – based on its direct derivation from architecture management literature – as well as the compliance of the consolidated process model regarding the list of requirements.

As a result of the paper, it can be stated that evolutionary, process-oriented application architecture management is comprised of four phases (= interrelated sub-processes): architecture planning, architecture development, architecture communication and architecture lobbying. The cases exhibit compatibility not only regarding these four sub-processes, but also similarities regarding the activities which make up these phases (cf. section 5)

Further research in the area of application architecture management is necessary in several directions. With respect to the developed process model, it will be necessary to analyze its "reference" character by means of additional case studies. On the other hand, an in-depth qualitative evaluation of the fundamental process models is required in order to give the consolidated model even more explicit to-be character [20] in the sense of reference modeling. Finally, the process model has to be further detailed by means of analysis / design technique specifications, and has to be supplemented by a role model and an information model in order to enhance its added value in the form of a comprehensive method [3].

References

[1] Birkhölzer, T. and J. Vaupel, IT-Architekturen - Planung, Integration, Wartung, VDE, Berlin, 2003.

[2] Braun, C. and R. Winter, "A Comprehensive Enterprise Architecture Metamodel and Its Implementation Using a Metamodeling Platform", in: Enterprise Modelling and Information Systems Architectures, Proc. of the Workshop in Klagenfurt, Klagenfurt, 2005, pp. 64-79.

[3] Braun, C., F. Wortmann, M. Hafner and R. Winter, "Method Construction – A Core Approach to Organizational Engineering", in: Applied Computing 2005, Proc. of the 2005 ACM Symposion on Applied Computing, New York, NY, USA, 2005, pp. 1295-1299.

[4] CIO-Council, Federal Enterprise Architecture Framework Version 1.1, September 1999.

[5] Dern, G., Management von IT-Architekturen - Informationssysteme im Fokus von Architekturplanung und -entwicklung, Vieweg, Wiesbaden, 2003.

[6] Dietzsch, A., "Positionierung eines Unternehmensarchitektur-Ansatzes: Erfahrung der Schweizerischen Mobiliar im Architekturmanagement", in: Enterprise Architecture und Enterprise Application Integration (EAI) - Proceedings of the GI-Arbeitskreis Enterprise Architecture Frühjahrskonferenz 2003, St. Gallen, 2003, pp. 50-61.

[7] Gutzwiller, T., Das CC RIM-Referenzmodell für den Entwurf von betrieblichen, transaktionsorientierten Informationssystemen, Physica, Heidelberg, 1994.

[8] Hafner, M., Entwicklung einer Methode für das Management der Informationssystemarchitektur im Unternehmen, PhD Thesis, Institute of Information Management, University of St. Gallen, St. Gallen, 2005.

[9] ISO, Stages for the Development of International Standards, Electronic Resource, http://www.iso.org/iso/en/stdsdevelopment/whowhenhow/proc/proc.html, Last Accessed 2007-06-13.

[10] ISO/IEC, ISO/IEC Directives, Part 2: Rules for the structure and drafting of International Standards, Electronic Resource, International Organization for Standardization, International Electrotechnical Commission.

[11] James, G., Seven Architecture Management Best Practices, Electronic Resource, Gartner Commentary 18-9663, http://poseidon.iwi3.unisg.ch/gartner/content/research/112300/112330/112330.html, Last Accessed 2003-08-13.

[12] Krüger, S. and J. Seelmann-Eggebert, IT-Architektur-Engineering - Systemkomplexität bewältigen, Kosten senken, Potenziale freisetzen, Galileo Press, Bonn, 2003.

[13] Mesarovic, M.D., D. Macko and Y. Takahara, Theory of Hierarchical, Multilevel Systems, Academic Press, New York, London, 1970.

[14] Nuseibeh, B. and S. Easterbrook, "The Process of Inconsistency Management: A framework for understanding", in: Proc. of the 10th Internat. Workshop Database Expert System Applications, Florence, Italy, 1999, pp. 364-368.

[15] Opengroup, TOGAF "Enterprise Edition" Version 8.1, Electronic Resource, The Open Group, http://www.opengroup.org/architecture/togaf8-doc/arch/, Last Accessed 2007-06-13.

[16] Österle, H., Business in the Information Age - Heading for New Processes, Springer, New York, 1995.

[17] Österle, H., W. Brenner and K. Hilbers, Unternehmensführung und Informationssystem - Der Ansatz des St. Galler Informationssystem-Managements, Teubner, Stuttgart, 1992.

[18] Paulish, D.J., Architecture-Centric Software Project Management, Addison-Wesley, Upper Saddle River, NJ (USA), 2002.

[19] Perks, C. and T. Beveridge, Guide to Enterprise IT Architecture, Springer, New York, 2003.

[20] Rosemann, M. and R. Schütte, "Grundsätze ordnungsmäßiger Referenzmodellierung", in: Entwicklungsstand und Entwicklungsperspektiven der Referenzmodellierung, Proceedings zur Veranstaltung vom 10. März 1997, Münster, 1997, pp. 16-33.

[21] Schütte, R., Grundsätze ordnungsmässiger Referenzmodellierung, Gabler, Wiesbaden, 1998.

[22] vom Brocke, J., Referenzmodellierung – Gestaltung und Verteilung von Konstruktionsprozessen, Logos, Berlin, 2003.

[23] Werres, M., Enterprise Architecture Mangement: The IBM Approach, Electronic Resource, http://akea.iwi.unisg.ch/downloads/20020909-ibm.pdf, Last Accessed 2007-06-13.

[24] Winter, R., "Architektur braucht Management", Wirtschaftsinformatik, Vol. 46, No. 4, 2004, pp. 317-319.

[25] Wortmann, F., Entwicklung einer Methode für die unternehmensweite Autorisierung, PhD Thesis, Institute of Information Management, University of St. Gallen, St. Gallen, 2005.

[26] Zachman, J.A. and J.F. Sowa, "Extending and formalizing the framework for information systems architecture", IBM Systems Journal, Vol. 31, No. 3, 1992, pp. 590-616.