Something Is Missing: Enterprise Architecture from a Systems Theory Perspective

Sebastian Kloeckner and Dominik Birkmeier

University of Augsburg, Universitaetsstr. 16, 86159 Augsburg
{sebastian.kloeckner,dominik.birkmeier}@wiwi.uni-augsburg.de

Abstract. Enterprise modeling has been an area of significant research in the information systems discipline throughout the last decade. Mainly developed by IT-practitioners, enterprise architectures (EA) became a promising and comprehensive approach to model either the current or desired state of enterprises. Existing approaches are, however, often criticized for paying too little attention to the business side of enterprises. In this paper, we interpret an enterprise as socio-technical system and analyze from a systems theory perspective which features are necessary for a comprehensive model. From there, we deduce if, why and how additional aspects of enterprises should be included into EA. Amongst others, it becomes obvious that especially human actors, as most flexible and agile elements of enterprises, are not adequately included in current architectures. We therefore present first ideas for integrating this important aspect into EA, the corresponding implications of such an inclusion, as well as several areas of further research.

Keywords: enterprise architecture, systems theory, socio-technical systems.

1 Introduction

Enterprise modeling in general and enterprise architectures (EA) in particular have gained significant momentum in research (e.g. [1, 2]) and practice (e.g. [3]) during the last years. Especially the promise to make the important elements of an enterprise and their relations visible makes it an interesting concept for the analysis and design of complex business systems. A comprehensive enterprise architecture therefore specifies, amongst others, the goals and strategies of an enterprise, its business processes as well as the associated resources like production systems, information systems and humans [4]. While the former aspects are often included in current concepts of EA, especially humans, as integral parts of enterprises, are often not taken into consideration. But only such a complete picture would essentially support necessary transformations of organizations in a flexible and agile way.

First being discussed in the 1970s, several enterprise architecture concepts, often under different names, were proposed over the time (e.g. [2, 5-7]). Today the Zachmann-Framework [8], The Open Group Architecture Framework [9] and the Federal Enterprise Architecture [10] can be seen as the most widespread frameworks for enterprise architectures. But as most of these concepts are rooted in the IT-departments of today's enterprises, they are oftentimes strongly focused on IT-related aspects.

A. Dan, F. Gittler, and F. Toumani (Eds.): ICSOC/ServiceWave 2009, LNCS 6275, pp. 22–34, 2010.

Although business issues are slowly moving into research focus [11, 12], few attentions have so far been paid by the information systems science community to organizational aspects. When looking at the complete picture, enterprises are socio-technical systems and therefore do not only contain technical components, but also humans and the organizational context [13]. Especially this organizational context can have a significant impact on the overall success and is one of the main reasons of budget overruns or even complete failings, as stakeholders are resistant to change or do not adopt new technologies. Dietz and Hoogervorst [14] consider the traditional *black-box* thinking based knowledge, i.e., knowledge about the function and the behavior of enterprises, as one of the reasons for such problems. While being sufficient for managing an enterprise within its current range of control, this kind of thinking is inadequate when an enterprise has to change. For such cases, a *white-box* approach, describing the construction and operation of enterprises, is needed.

In this paper, we therefore take an argumentative-deductive approach to analyze from the holistic perspective of systems theory, in particular Ropohl's *Theory of General Technology* [15], where certain aspects are missing in the current concept of enterprise architectures. Furthermore, we present first ideas how these missing parts can be integrated and which benefits could be realized by such an integration.

In the remainder of the paper we will proceed as follows: after presenting related work and motivating the research gap in the next section, we introduce relevant theories and concepts. In detail, we present the common understanding of enterprise in the field of enterprise architectures, as well as the basic concepts of systems theory in general and the Theory of General Technology in particular. By interpreting enterprises as socio-technical systems, different aspects in regard to including human factors into enterprise architecture are deduced in the synthesis. Taking these aspects into account we draw the conclusion that especially an integration of human beings into the lower layers of EA is necessary. In the following section on implications for practice and academia we present that this will further disclose new optimization approaches and cost-saving opportunities. Finally, we sum up our findings, provide an outlook and raise several possible trends for EA.

2 Related Work

The evaluation and comparison of EA frameworks in general and their completeness in particular has frequently been addressed in literature. Besides publications focusing on a single framework (e.g. [16]) there are also several extensive comparisons between existing frameworks such as the ones from Leist et al. [17], Bernus et al. [18], and Schekkerman [19]. All of them elaborate on the individual strengths and weaknesses of the common frameworks and try to identify their gaps as well as possibilities for further improvement. Noran [20] furthermore examines the mapping of classical frameworks onto the Generalized Reference Architecture and Methodology (GERAM) Framework [12], which tries to provide a common, but rather abstract, regulation framework for the former ones.

In addition to those comparisons between frameworks, Aier et al. [21] provide a literature survey on established contributions from academia and practice, as well as an empirical examination on comprehension, model building and utilization of enterprise architectures. They develop a systematic overview on the current understanding and the state of the art of EA. From there, they identify discrepancies between research and practice and discuss corresponding implications for both. Through an empirical survey among practitioners and researchers in the area of EA, they identify, amongst others, which design artifacts have to be considered in an EA and the degree of their realization in practice. From this survey, it can be seen that the interaction with customers and suppliers, roles and responsibilities, as well as organizational units are all considered mainly important, but their degree of realization is generally lower than the average of all design artifacts. On the other side, many of the aspects considered as the most important ones, with the highest degree of realization, are purely technical. Those are, amongst others, interfaces, applications, data structures, software-, hardware- and network-components, etc. It can be seen from the study that, while organizational aspects of enterprises slowly come into focus, enterprise architectures are predominantly shaped by IT-departments and their view of the enterprise.

Furthermore, an analysis of the "impact of service-oriented architecture on enterprise systems" was carried out by Bieberstein et al. [22]. Based on it, Schroth [23] presented his view of a Service-Oriented Enterprise. Both elaborate on the alignment of service-oriented architectures (SOA) with first aspects of existing organizational theories. Bieberstein et al. state that "the SOA paradigm also needs to be extended to transmute organizational structures and behavioral practices" [22] and they thus "propose the Human Services Bus (HSB), a new organizational structure that optimizes the workflow and streamlines cross-unit processes to leverage the new IT systems" [22]. Schroth [23] proposes an approach of mapping the major underlying principles of SOA, namely decentralization, agility, as well as composition and coordination of building blocks, to upcoming forms of organizations. He uses the HSB to allow advertising human services within an enterprise and provide a means for workflow monitoring. While both works consider aspects from organizational theory, they origin from a rather technical SOA context and focus on changes that an integration of SOA implicates for enterprises and so do not extend their findings to a general EA.

In summary it can be said that the issue of evaluating EA approaches has been faced from different perspectives. An extensive analysis and comparison of current frameworks, as well as a critical reflection of differences in research and practice has been utilized to identify goals for the future development. On the other hand, the idea of including aspects from organizational theory has been raised, but was limited to the field of service-oriented architectures. Though, to the best knowledge of the authors there is so far no comprehensive analysis of enterprise architecture in regard to systems theory from a scientific perspective.

3 Conceptual Foundations

As the review of the related work has illustrated, current concepts of EA are often focused on IT-related aspects. While understandable as most concepts have their origins in IT-departments, the call for a scientific backing of these concepts and an inclusion of organizational aspects gets louder. In order to get an insight into existing concepts and possible scientific theories, we firstly present the current understanding of enterprise architecture and the basic elements of systems theory.

3.1 Enterprise Architecture

As shown before, enterprise modeling is a field of significant research and is widely accepted in science and practice [4, 11]. Over time several different names, perspectives and definitions for enterprise architectures were proposed (e.g. [2, 5-7]). A comprehensive overview of different enterprise modeling approaches with sometimes different perspectives can be found in [21]. For a common understanding of the used terms, we will rely on the following definitions: In the context of EA, The Open Group defines an enterprise as "Any collection of organizations that has a common set of goals and/or a single bottom line. In that sense, an enterprise can be a government agency, a whole corporation, a single department, or a chain of geographically distant organizations linked together by common ownership" [9]. On the other hand, the ANSI/IEEE Standard 1471-2000 defines architecture 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" [24].

While most of the available modeling approaches conform to these definitions, they diverge in certain aspects due to their respective goals and complexity. Winter and Fischer criticize that "Traditionally, architecture in the information systems is focusing on IT related artifacts […]" and suggest that "Only when 'purely' business related artifacts are covered by EA, important management activities like business continuity planning, change impact analysis, risk analysis and compliance can be supported." [11]. Based on a literature analysis, they deduce five essential layers and related artifacts of enterprise architectures, which were adapted by Aier, Riege and Winter [21] and are shown in figure 1. These proposed layers are a first step to incorporate organizational aspects into the concept of enterprise architecture, as they include artifacts like organizational goals, organizational units and responsibilities. However, they still do not consider these aspects on the lower layers. When taking the history of enterprise architecture and its roots in the IT-departments of enterprises into account, it becomes understandable that it is, mainly on the lower layers, still strongly focused on IT-related issues. But, this perspective only examines one side of the coin, as it ignores the human contribution to the overall performance.

Going further back into history, it becomes obvious that enterprises and their architectures existed before IT came into play. At this time information systems, in their classical definition as systems for the communication and processing of information, without any or only little technology (letter shoot, dockets, etc.) were already in place. The most important components of such systems and the corresponding architectures were, and probably are, human beings. While literature often states flexibility and agility as primary goals of enterprise architectures (e.g. [25]), many approaches do not take these extraordinary flexible and agile "components" of an enterprise into account. Again, this might be owed to the fact that, until recently, practitioners of IT-departments have been leading in the development of the EA discipline. It therefore seems to be advisable to examine from a scientific perspective *if*, *why* and *how* these important providers of services and functionality should be included into enterprise architecture in general. And as enterprises are often described as systems, the holistic perspective of systems theory in combination with the concept of socio-technical systems seems to be a good starting point for such an examination.

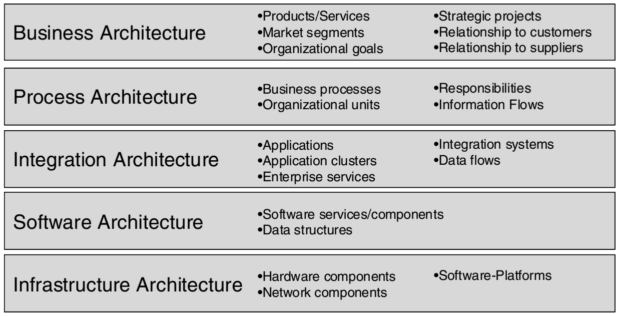

Fig. 1. Proposed layers and artifacts of Enterprise Architecture [21]

3.2 Socio-technical Systems

Systems theory is an interdisciplinary theory, which focuses on the description and analysis of any system as an integral whole that consists of interconnected elements (parts) and their relations. The roots of modern systems theory can be traced to Bertalanffy's General Systems Theory [26]. Mainly based on these works, Ropohl developed a Theory of General Technology [15] for explaining and analyzing elements and relationships of socio-technical systems. Socio-technical in this context refers to the interrelatedness of social and technical aspects of a system, e.g. human beings and technical artifacts in an organization. In his depictions, Ropohl first distinguishes between three different system perspectives: the functional, the structural and the hierarchical view. The functional system concept regards the system as a black-box, which is characterized by certain relations between in- and output properties that can be observed from outside. The structural system concept views the system as a whole, consisting of interrelated elements. And the hierarchical system concept finally emphasizes the fact that parts of a system can be considered as systems themselves and that the system itself is part of a larger system.

Based on these definitions, Ropohl develops a general model of an action system, which is characterized by three sub systems: the goal setting system (GS), the information system (IS) and the execution system (ES). The ES obtains material and energetic attributes and performs the basic work. In the context of business information systems, the mentioned material has to be interpreted as data and information, which have to be processed. The IS handles informational attributes. It absorbs, processes, and passes information and deduces instructions for the execution system from this information. The GS creates the system's internal goals as maxim of action. By introducing the concept of division of labor, Ropohl then further decomposes the initially "monolithic" system from a different perspective. As each action can consist of more than one sub-action, united in one virtual or real system, these sub-actions can also be disassembled into own systems. These new subsystems then have to be linked and coordinated in order to fulfill the formerly united action. The combination of the functional decomposition of a single action and the division of labor (chain of action, resp. workflow) is shown in figure 2.

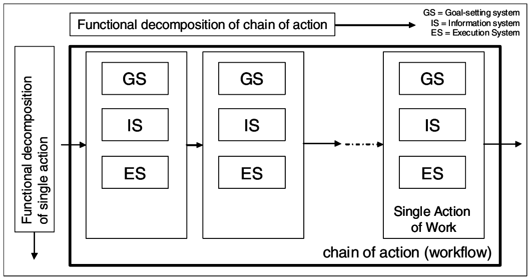

Fig. 2. Functional decomposition of single action and chain of action

Based on these two perspectives of decomposition, Ropohl then develops the concept of socio-technical division of work, shown in figure 3. Ropohl considers Adam Smith [27] to be the first, who said, while in a different context, that machines are similar to humans in many ways and that machines are small systems, created to cause certain movement effects.

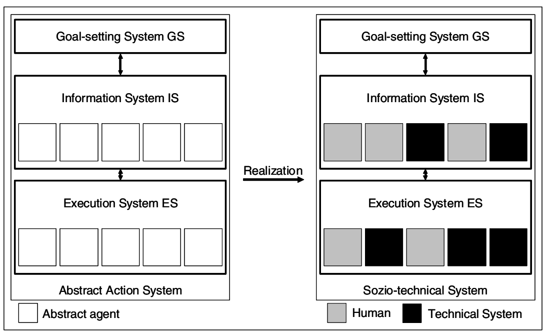

Fig. 3. Socio-technical division of work (adapted from [15])

The abstract action system, depicted on the left side of figure 3, contains an abstract agent as the agent of action. This system can be interpreted as a human being which incorporates all subsystems or as an organization, where several actors are in charge of certain sub-actions. When interpreted as an organization and transferred into reality the single sub-actions of the respective subsystems can be either assigned to human actors or to technical artifacts. Therefore, a socio-technical system is defined as an action or work system, which is composed of human beings and technical systems (artifacts). But while technical artifacts have clearly defined system boundaries, human individuals have additional properties, exceeding their predefined roles in the system, due to their social and cognitive capabilities (e.g. social networks, creativity, problem solving, etc.).

4 Synthesis

A comparison of the basic structures of enterprises and the theory of socio-technical systems shows that many similarities exist. An organization composed of humans and artifacts, like an enterprise, must therefore be interpreted as a socio-technical system, whose structure, as stated in ANSI/IEEE Standard 1471-2000 [24], is described by its architecture. Its actions are realized by humans and/or technical artifacts and from an abstract point of view, the different management and execution layers of enterprises can directly be mapped to the different subsystems of the functional decomposition: the goal-setting system is realized by the executive managers, the information system by the middle managers and/or information systems and the execution systems is materialized by the workers and/or production systems. Due to the recursive structure of systems theory, each of these three subsystems can again be interpreted as system and therefore consists of the three subsystems itself, which are realized by humans and/or artifacts.

Looking at the previously described layers and artifacts of EA, many of the proposed elements also fit with the concept of socio-technical systems: The business architecture with its goals, strategic projects, desired products, targeted market segments, etc., models the goal-setting system of the enterprise. The process architecture, with its business processes, information flows and responsibilities, etc., as well as the integration architecture, containing applications, enterprise services, integration systems, data flows, etc., mirrors the idea of the information system. The software and infrastructure architectures can finally be mapped onto the execution system.

As shown, there are obvious similarities between EA and socio-technical systems and as enterprises are often characterized as socio-technical systems, it seems to be reasonable to further analyze the concept of enterprise architecture under the perspective of the Theory of General Technology. Especially, as it also becomes evident that the respectively considered elements, subsystems and relations seem to be different.

Considering the literature on EA, it first becomes apparent that EA models are used for two different purposes: Many publications about enterprise architectures differentiate between descriptive and design models (e.g. [3, 4, 16]). Descriptive models, often called “As-Is-Models”, illustrate the state of an enterprise in its current situation and how business is executed. Design models, regularly labeled as “To-Be-Models”, envision a future state of the enterprise and how it should be shaped in the future. As the purpose of these models significantly differs, it becomes obvious that they are in fact the outcome of two different types of systems. The first system, which has the descriptive model as result, can be named *run-time system*, the second, producing the design model, should be called *design system*. These two systems are not independent of each other, though. The design system pictures a future state, which shall be implemented in the run-time system. The existing run-time system on the other side is usually the basis for a new design, as complete green field approaches are typically not possible in practice. These two systems themselves contain socio-technical sub-systems. Each of these subsystems realizes certain actions, which can or cannot be fulfilled by humans and/or technical artifacts as shown in table 1. In addition, each type of agent (human being and technical artifact) has distinct properties, which make them better suitable for a certain task.

During design time, the goal-setting-subsystem can only be realized by humans, since the creation of goals, targets, decisions and strategies needs creativity, which cannot be realized in technical artifacts. While a support by technical artifacts is possible, e.g. decision support systems, a replacement of humans by technical artifacts is not possible. The information subsystem of the design system has to define the necessary flows and tasks in order to make a realization of the defined goals and targets possible.

While first approaches are made to at least partly automate the action of this subsystem (e.g. [28]), it is unlikely that it can be fully automated in the near future. Finally, the execution subsystem of the design system creates (make) or acquires (buy) the necessary resources to fulfill the tasks defined by the information subsystem. While this action is usually realized by humans, several approaches, like code generators or automated market places, try to computerize it.

Table 1. Different actions of sub systems at design and execution time

	Design System		Run-Time System	
	Action	**Agent**	**Action**	**Agent**
GS	(Re-)Define targets of system and actions	Humans only	Control and/or set goal of executed action	Humans and technical artifacts (goals are implicitly set)
IS	(Re-)Define necessary flows and tasks to achieve goals	Mostly humans, sometimes technical artifacts	Execute control flow or function templates, call necessary functions	Humans and technical artifacts
ES	Create and acquire necessary flow and function templates	Humans and technical artifacts	Realize and execute certain function	Humans and technical artifacts

The goal-setting subsystem of the run-time system can be realized by humans or by technical artifacts. In the case of humans, they will follow the plans, which were defined during design time. If certain situations were not considered during design time, they are able to (re-)define what has to be done and how to react to certain influences from the environments in a flexible as well as agile way. For technical artifacts the goal-setting subsystem exists only implicitly. By defining its functionality during design time, its goals are irrevocably set. If certain system-external or -internal influences or states were not considered at design time, e.g. by catching errors, a technical artifact will malfunction, e.g. an error occurs. An independent redefinition of goals and a corresponding reaction is, while being researched, not possible yet. According to the defined goals, the information subsystem executes the created or acquired flow and function templates. It controls, depending on the input and based on these templates, which agents, humans or technical artifacts, have to be called to fulfill a certain action. While humans can independently switch to comparable functions if a needed subsystem is not available (e.g. different supplier of comparable pre-products), this is, due to the normally missing capability of comparing unspecified possibilities, usually not possible for technical artifacts. The best chance for technical artifacts, if considered during design time, is to keep state until the required subsystem becomes available. The execution subsystem finally executes the defined tasks. It takes the input and transfers it into the output. The actions of the execution subsystem can, due to the hierarchical concept of systems theory, themselves be complex systems. They can be realized by an enterprise, like a sub-contractor, by technical artifacts, like web services or by human beings. Now, taking into account that the results of the information subsystem of the design system are used by the goal-setting subsystem of the run-time system and that the outcomes of the execution subsystem at design time are utilized by the information subsystem at run time, the different subsystems can be connected as shown in table 2. In addition, it can again be shown that the different subsystems can also be mapped onto the proposed layers of EA.

Table 2. Mapping of socio-technical subsystems to proposed layers of enterprise architecture

Layers of EA	Design System	Run-Time System
Business Architecture	(Re-)Define targets of system and actions	---
Process Architecture	(Re-)Define necessary flows and tasks to achieve goals	Control and/or set goal of executed action
Integration Architecture	Create and acquire necessary flow and function templates	Execute control flow or function templates, call necessary functions
Software Architecture	---	Realize and execute certain function
Infrastructure Architecture	---	---

5 Consequences of the Synthesis

Taking this into account, several consequences for and gaps of enterprise architecture can be deduced. While the higher layers still contain the human aspect of socio-technical systems, they disappear on the lower layers like integration, software and infrastructure architecture. Although it is understandable for the infrastructure level, where offices and buildings, walks and streets would correspond to hardware and network components, it has to be questioned, if this is comprehensible for the integration and software layer. When looking into today's enterprises, human beings regularly execute functions of these layers: For example, whenever a direct information system integration is not possible, due to technical restrictions, human beings or teams connect the two distinct application systems over their graphical user interfaces and act as human connectors between these systems. But, as these human connectors are usually very costly compared to technical interfaces, information about them and their integration into the enterprise architecture would help to optimize the enterprise. In this context, human teams could be interpreted as enterprise services which offer certain business and integration functionality to other elements of the system.

When looking at the layer of software architecture with its software components and data structures, again, the general tasks are sometimes carried out by human beings. E.g., humans often carry specialized data about certain customer preferences or realize tasks which require a certain portion of creativity. Both are characteristics which are important for successful enterprises and cannot easily be realized by technical artifacts. When these qualities are overseen, just because they were not part of the enterprise architecture, for example when replacing sales clerks by web front ends or call centers, the overall performance of the enterprise can take severe damage. In this context, human actors could be understood as software or elementary services which carry data or realize certain specialized functionality. The breakdown or loss of such system elements and the corresponding implications can be compared with those of software services, although the latter are often easier to fix.

It therefore seems to be more than reasonable to integrate human beings or aggregations of those (e.g. teams) as components or service providers with certain interfaces into the lower layers of enterprise architecture, as shown in figure 4. Only if human beings, as integral part of the socio-technical system "enterprise" are included, profound management and alignment decisions become possible.

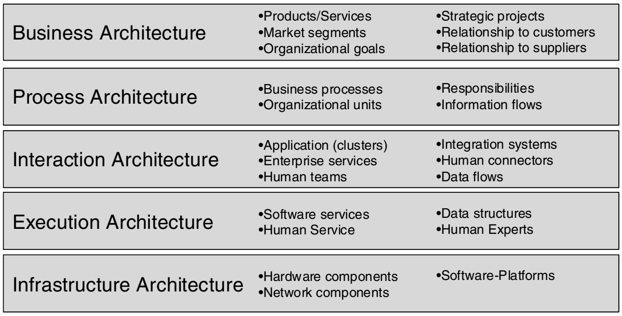

Fig. 4. Integration of human factors into proposed layers and artifacts of EA

In consequence and in order to mirror the inclusion of human beings and technical artifacts properly, the name of the integration and software architecture should be changed: Instead of using the word *integration*, the word *interaction* would better reflect the interplay between humans and technical artifacts. And *execution* architecture better characterizes the realization of basic services, regardless of whether they are provided by humans or technical artifacts.

6 Implications for EA in Practice and Research

The integration of the human or social aspect into enterprise architecture has several implications for the successful management of enterprises and the focus of future research. First, it would allow the concept of enterprise architecture to become a comprehensive model of all relevant aspects and elements of the enterprise. The white-box perspective of systems theory has shown that each subsystem of the supersystem enterprise is usually a socio-technical system and omitting one important element of this system would be negligent and carries the risk of wrong decisions.

Second, as an increase of agility (adapt to unexpected changes) and flexibility (adapt to expected changes) is one of the primary objectives of the concept of enterprise architecture [25], it seems to be reasonable to include the most agile and flexible components of the socio-technical system enterprise, human beings and organizational groups, into the concept. Especially due to their flexible goal-setting and information system, human beings can quickly adapt to unexpected as well as to expected changes of the environment. A comparably quick adaption of technical artifacts has to be viewed as unlikely. Furthermore, an a priori integration of solutions for unexpected changes is impossible, as they can, by definition, not be foreseen. Therefore, the inclusion of human beings into the concept of enterprise architecture would significantly increase the level of usable resources.

Third, the inclusion of human aspects into enterprise architecture would be the first step to the often demanded integration of concepts from organizational science (e.g. [13]). In addition, an integration of human and organizational aspects would reveal areas, where organizational science or information systems science tries to solve comparable questions, which were already answered by the neighbor discipline. For example, while the component and service identification community (e.g. [29, 30]) still tries to find the best-suited solution for the grouping of technical artifacts (granularity), the organizational science already gave answers to these questions in the early 1970s (e.g. [31, 32]). Under the name "departmentalization" the authors utilize affinity analysis to create units with maximal cohesion and minimal dependencies.

Forth, if humans are considered as part of the system, the role of information system technology can change: Instead of information technology being part of human-controlled systems, humans can become part of technical-controlled systems, e.g. in the case where the process flow (information system of the run-time system) is controlled by the IT-system and humans fulfill certain tasks as part of the execution system (at execution time). While this will be bought at the cost of an asynchronous execution, as humans are slower than usual time-outs of IT-systems, it will offer new functionality and realization potentials. And perhaps this shift in perspective could even result in a change of existing programming paradigms.

Finally, only the integration of humans into the concept of enterprise architecture will shed light on spots, where new technical artifacts would allow significant cost reductions. Based on activity-based costing [33] for formerly manual tasks, the amount of cost-savings can be exactly calculated and used as argument for additional investments in information technology. On the other side, there may also occur situations, where technical artifacts should be replaced by humans; for example due to their higher agility and flexibility or when the total costs of ownership (TCO) exceed the costs of labor. In order to make such decisions, a new measure like *Total cost of IT usage*, as an aggregate of TCO and costs of labor, might become reasonable.

7 Conclusions and Further Research

In this paper we analyzed the concepts of enterprise architecture from a systems theory perspective. By interpreting enterprises as socio-technical systems and under usage of the Theory of General Technology, we deduced that the concept and use of enterprise architectures implicitly includes two different systems: The design system and the run-time system. These two systems fulfill different purposes and therefore require different capabilities. Due to these requirements, it becomes obvious that humans and groupings of them play an important role in today's enterprises and the negligence of their actions and properties results in incomplete models. Especially, as humans are the only components in the system "enterprise", which are able to act in an agile way. While we did not explicitly name service-oriented architectures, the interpretation of humans as service providers would also perfectly integrate into the paradigm of SOA and e.g. in fact supports the realization of a Human Service Bus [22]. But the inclusion of the human factor into the concept of EA also creates several open questions:

- What shift in perspective is necessary, if humans, as active and passive elements of enterprises, in opposition to technical artifacts, as solely passive components, are included into enterprise architectures?
- How can humans and technical artifacts be included into identification approaches?
- How can additional capabilities of human individuals (e.g. social networks), not directly connected with their role in the organization, be reflected in models?
- What side-effects are created by the inclusion of the human factor?

- What measures are necessary to decide if the replacement of humans by technical artifacts or vice versa is reasonable?
- Do situations exist, where the replacement of technical artifacts by humans is reasonable and how can the proportion between humans and technical artifacts be optimized?

When further applying the white-box perspective of systems theory, many other questions could also be taken into consideration. However, already the inclusion of human actors into EA models will shed light onto those actually unobserved areas of the socio-technical system enterprise. In order to answer these questions, further research, especially from a holistic, scientific perspective is needed. Only if the needs of today's practice and the systematic procedures of science are combined, really comprehensive models and methods can be created.

References

1. Arbab, F., Boer, F.d., Bonsangue, M., Lankhorst, M., Proper, E., Torre, L.v.d.: Integrating Architectural Models - Symbolic, Semantic and Subjective Models in Enterprise Architecture. Enterprise Modelling and Information Systems Architectures 2 (2007)
2. Frank, U.: Multi-perspective Enterprise Modeling (MEMO) - Conceptual Framework and Modeling Languages. In: HICSS 2002, vol. 3, p. 72. IEEE Computer Society, Hawaii (2002)
3. Gaertner, W.: Ansatz für eine erfolgreiche Enterprise Architecture im Bereich Global Banking Division/Global Transaction Banking IT and Operations der Deutschen Bank. Wirtschaftsinformatik 46, 311–313 (2004)
4. Sinz, E.: Unternehmensarchitekturen in der Praxis – Architekturdesign am Reißbrett vs. situationsbedingte Realisierung von Informationssystemen. Wirtschaftsinformatik 46, 315–316 (2004)
5. Scheer, A.-W., Schneider, K.: ARIS — Architecture of Integrated Information Systems. In: Bernus, P., Mertins, K., Schmidt, G. (eds.) Handbook on Architectures of Information Systems, pp. 605–623. Springer, Heidelberg (2006)
6. Johnson, P., Ekstedt, M.: Enterprise Architecture: Models and Analyses for Information Systems Decision Making. Studentlitteratur, Pozkal (2007)
7. Ferstl, O.K., Sinz, E.J.: Modeling of Business Systems Using SOM. In: Bernus, P., Mertins, K., Schmidt, G. (eds.) Handbook on Architectures of Information Systems, pp. 347–367. Springer, Heidelberg (2006)
8. Zachman, J.A.: A framework for information systems architecture. IBM Systems Journal 26, 277–293 (1987)
9. The Open Group: TOGAF "Enterprise Edition" Version 8.1 (2002)
10. United States Office of Management and Budget: FEA Consolidated Reference Model Document Version 2.3 (2007)
11. Winter, R., Fischer, R.: Essential Layers, Artifacts, and Dependencies of Enterprise Architecture. Journal of Enterprise Architecture (2007)
12. IFIPIFAC Task Force: GERAM: Generalised Enterprise Reference Architecture and Methodology Version 1.6.3 (1999)
13. Picot, A., Baumann, O.: The Relevance of Organisation Theory to the Field of Business and Information Systems Engineering. Business & Information Systems Engineering 1 (2009)

14. Dietz, J.L.G., Hoogervorst, J.A.P.: Enterprise ontology in enterprise engineering. In: ACM Symposium on Applied Computing. ACM, Fortaleza (2008)
15. Ropohl, G.: Allgemeine Technologie. Eine Systemtheorie der Technik, Hanser, München/Wien (1999)
16. Noran, O.: An analysis of the Zachman framework for enterprise architecture from the GERAM perspective. Annual Reviews in Control 27, 163–183 (2003)
17. Leist, S., Zellner, G.: Evaluation of current architecture frameworks. In: 2006 ACM symposium on Applied computing, Dijon, France, pp. 1546–1553 (2006)
18. Bernus, P., Nemes, L., Schmidt, G. (eds.): Handbook on Enterprise Architecture. Springer, Heidelberg (2003)
19. Schekkerman, J.: How to Survive in the Jungle of Enterprise Architecture Frameworks: Creating or Choosing an Enterprise Architecture Framework, Trafford, Victoria, BC (2004)
20. Noran, O.: A Meta-Methodology for Collaborative Networked Organisations. Griffith University, Brisbane (2004)
21. Aier, S., Riege, C., Winter, R.: Unternehmensarchitektur – Literaturüberblick und Stand der Praxis. Wirtschaftsinformatik 50, 292–304 (2008)
22. Bieberstein, N., Bose, S., Walker, L., Lynch, A.: Impact of service-oriented architecture on enterprise systems, organizational structures, and individuals. IBM Systems Journal 44, 691–708 (2005)
23. Schroth, C.: The Service-Oriented Enterprise. In: TEAR 2007, St. Gallen, Switzerland (2007)
24. IEEE Computer Society: Recommended Practice for Architectural Description of Software-Intensive Systems. IEEE Std 1471-2000, New York (2000)
25. Schelp, J., Aier, S.: SOA and EA - Sustainable Contributions for Increasing Corporate Agility. In: HICSS 2009. IEEE Computer Society, Waikoloa (2009)
26. Bertalanffy, K.L.v.: General System theory: Foundations, Development, Applications. George Braziller, New York (1976)
27. Smith, A.: An inquiry into the nature and causes of the wealth of nations (1776)
28. Heinrich, B., Bewernik, M.-A., Henneberger, M., Krammer, A., Lautenbacher, F.: SEMPA – Ein Ansatz des Semantischen Prozessmanagements zur Planung von Prozessmodellen. Wirtschaftsinformatik 50, 445–460 (2008)
29. Albani, A., Overhage, S., Birkmeier, D.: Towards a Systematic Method for Identifying Business Components. In: Chaudron, M.R.V., Szyperski, C., Reussner, R. (eds.) CBSE 2008. LNCS, vol. 5282, pp. 262–277. Springer, Heidelberg (2008)
30. Aier, S., Winter, R.: Virtuelle Entkopplung von fachlichen und IT-Strukturen für das IT/Business Alignment – Grundlagen, Architekturgestaltung und Umsetzung am Beispiel der Domänenbildung. Wirtschaftsinformatik 51 (2009)
31. Müller-Merbach, H.: OR-Ansätze zur optimalen Abteilungsgliederung in Institutionen. In: Kirsch, W. (ed.) Unternehmensführung und Organisation, Wiesbaden, pp. 93–124 (1973)
32. Kieser, A.: Abteilungsbildung. In: Frese, E. (ed.) Handwörterbuch der Organisation, Poeschel, Stuttgart, vol. 3, pp. 57–72 (1992)
33. Peacock, E., Tanniru, M.: Activity-based justification of IT investments. Information and Management 42, 415–424 (2005)