

# Towards a Principle-Based Framework for Enterprise Architecture Rationalization

Diana Marosin<sup>1,2,3</sup>

<sup>1</sup> CRP Henri Tudor, Luxembourg, Luxembourg

<sup>2</sup> Radboud University Nijmegen, Nijmegen, the Netherlands

<sup>3</sup> Enterprise Engineering Team, Luxembourg, Luxembourg

diana.marosin@tudor.lu

**Abstract.** Organizations increasingly use enterprise architecture (EA) to direct change processes, given that it provides means to direct and steer their transformation and innovation, while at the same time ensuring the alignment between different key aspects (e.g. business services, business process, services, IT systems). In creating enterprise architectures, several design decisions have to be made. Such decisions are to a large extent based on assumptions about the situation at hand. The aim of this PhD research is to explicitly link underlying assumptions to architectural design decisions in order to make the rationalization of these decisions explicit as well as traceable in terms of formal reasoning. In this paper we present the state of the art on EA principles, the link between EA principles, EA decisions and computer science tools that can accommodate principle oriented decision making process. Furthermore, we present preliminary results and directions for future work.

**Keywords:** enterprise architecture, principles, assumptions, decision making, rationalization

## 1 Introduction

This work is part of the “RationalArchitecture” programme, started in the beginning of 2013, and sponsored by the “Fonds National de la Recherche Luxembourg” ([www.fnrl.lu](http://www.fnrl.lu)).

In practice, enterprises are confronted with frequent changes and challenges. The assumptions, and their relative priority, do not only depend on the specific stakeholders that are involved in creating the architecture, but also on the actual transformation. Therefore, it is desirable for these organizations to clearly trace any architecture related design decisions back to their assumptions. Having the ability to reason about the connection between architecture-level design decisions and their underlying assumptions, will (1) enable consistency checks of the architecture, (2) enable a more precise capturing of design knowledge, and (3) enable advanced impact/what-if analysis when confronted with changes of the underlying assumptions.

In the case of architectural decision-making, companies would ideally apply an a-priori rationalization. However, this is not well documented (if documented

at all). Therefore, being constrained by the existence of case studies, as well as the common practices, this project focus on a-posteriori rationalization. Our main research question is:

How to represent/capture the rationalization/motivation of architecture decisions?

We divide this question in the following sub questions:

1. How to position different types of assumptions, such as goals, requirements, principles? How to involve stakeholders and architects in validating them?
2. How to represent/capture the underlying assumptions? How to reason on the relation between assumptions and decisions?
3. How to reason on the relation between assumptions and decisions and associated uncertainties?
4. How to assess the robustness of design decisions in relation to the robustness of underlying assumptions?

The rest of the paper is structured as follows: in section 2 we summarize the existing architecture frameworks, position the existence of principles within them and summarize techniques present in computer science that could be used for reasoning on assumptions and architecture decisions. In section 3 we present the research methodology used for investigation. In section 4 we present preliminary steps taken in order to answer the research questions and section 5 concludes.

## 2 Related work

In this section we present non-exhaustively the field of enterprise architecture and the existing architecture frameworks. Furthermore, we summarize how architecture principles fit in these frameworks, how decisions are supported by architectural principles and how assumptions influence these principles. In the end of the section we introduce logical tools that we believe provide a good support for modelling decisions in an enterprise.

### 2.1 Enterprise architecture frameworks

Many definitions of enterprise architecture are used. We adhere to the following definition of architecture [6]. Furthermore, enterprise architecture is a specialisation of an architecture:

**Definition 1.** (Architecture) Those properties of an artifact that are necessary and sufficient to meet its essential requirements.

**Definition 2.** (Enterprise Architecture) The architecture of an enterprise. As such, it concerns those properties of an enterprise that are necessary and sufficient to meet its essential requirements.

Enterprise architecture is a comprehensive framework used to manage and align an organization. It describes the strategic direction of all composing elements of a company: organization structure, business processes, people and stakeholders, applications, data, infrastructure, etc. An architecture provides the guiding principles on how information and technology will support the business operations and provide benefit for the business. Therefore, enterprise architecture is about more than technology, it is about the entire organization (or enterprise) and identifies all of the pieces that make the organization work. The mission of enterprise architecture is to add value by providing management with means for informed governance of the enterprise transformation [20].

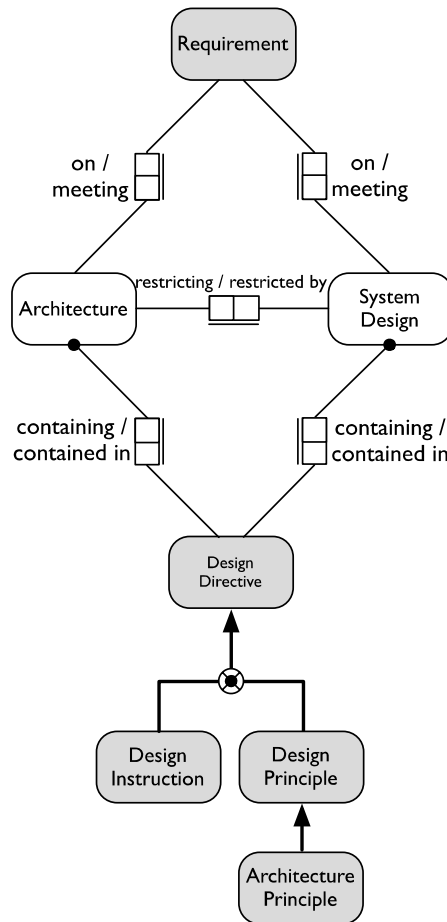
An architecture framework is a set of tools, methods and guidelines which can be used for developing a broad range of different architectures. It should describe a method for defining an information system in terms of a set of building blocks and show how they fit together. Also, it should contain a set of tools, provide a common vocabulary, include a list of recommended standards and a list of compliant products that can be used to implement the building blocks. The Open Group Architecture Framework (TOGAF) [29] is a standardized method for enterprise architecture. Large consultancy firms, such as IBM, HP, SAP and Capgemini have adopted it and enriched it with their own architectural knowledge and experience. TOGAF provides an elaborate reference on enterprise architecture, including an architecture development method, an architecture content framework, architecture reference models and an architecture capability framework.

The ArchiMate language, as described in [10], complements TOGAF, provides a vendor-independent set of concepts, including a graphical representation that helps creating a consistent, integrated model “below” the “waterline”, which can be depicted in the form of TOGAFs views. It has become the open standard for architecture modelling in the Netherlands, it is also fairly well known in the international enterprise architecture community, as it has been adopted by The Open Group. Resembling the Unified Modelling Language (UML), the ArchiMate modelling notation is intuitive and expressive enough to allow the modelling of all layers (business, application, and technology infrastructure) and all aspects (structure, behaviour, and information) of an organization in an integrated way.

## 2.2 Architecture principles

In practice, many different types of architecture principles are used. At the same time, principles are referred to by different names, including architecture principles, design principles, and IT policies. We use the definitions provided in [6]:

**Definition 3.** (Architecture principles) A design principle included in an architecture is a declarative statement that normatively prescribes a property of the design of an artifact, which is necessary to ensure that the artifact meets its essential requirements.



**Subtype defining rule:**

- An Architecture Principle is a Design Principle which is contained in some Architecture

Fig. 1: Conceptual framework for architecture principles [6]

In Figure 1 we present a meta-model incorporating architecture principles. **Design principles** are a declarative statements that normatively restrict the design freedom [7]. A normative principle is a declarative statement that prescribes a property of something (it can be seen as a “rule of conduct”). **Design instructions** are instructive statements that describe the design of an artifact. They contain usually concepts used in the actual construction of the enterprise (e.g., value exchange, transactions, services, contracts, processes) and use a representation language (e.g., UML, ArchiMate, BPMN, DEMO) [7]. Design instructions provide a more operational and tangible refinement of the design principles. Due their nature, design instructions allow an enterprise to simu-

late/analyse the effects of different options for the future, as well as analyse the current state and identify current problems [13]. In conclusion, **architecture principles** are design principles that focus on how the design of an enterprise will meet its essential requirements. They are declarative statements that can be made more precise using design instructions.

Formalising architecture principles comes from deep analysis of drivers (i.e., goals and objectives that stakeholders seek to meet) embedded in the strategy of the enterprise, the risks that may occur, potential opportunities, constraints.

TOGAF lists five criteria that distinguish a good set of principles: [1] **Understandable:** The underlying conviction can be quickly grasped and understood by individuals throughout the organization. [2] **Robust:** Each principle should be sufficiently definitive and precise to support consistent decision making in complex, potentially controversial, situations. [3] **Complete:** Every potentially important principle governing the management of information and technology for the organization is defined. [4] **Consistent:** Principles should not be contradictory to the point where adhering to one principle would violate the spirit of another. [5] **Stable:** Principles should be enduring, yet able to accommodate changes.

### 2.3 Formalising principles based on assumptions

Enterprise principles define the cornerstone of EA. There seems to be no universal agreement on the types of drivers that exist to motivate architecture principles [6]. The business motivation model [21] provides important concepts to express motivation. The model was initially created to provide the motivations behind business rules, but can also be used to find the motivation for architecture principles. This idea is brought to enterprise architecture by Engelsman *et al.* [5] who state that architecture principles are based on an assessment of stakeholder concerns. An assessment represents the outcome of the analysis of some concern, revealing the strengths, weaknesses, opportunities that may trigger a change to the enterprise architecture.

In this work one of our goal is to link the enterprise principles to their underlying assumptions. Major classes of assumptions include: [1] **Environment:** state and evolution of stakeholders in the enterprise; [2] **Enterprise:** strategic directions of the enterprise, goals of the stakeholders, business and IT strategies; [3] **Means:** several qualities of the components (to-be) used in the implementation of the enterprise.

General insights on the linkage between assumptions, stakeholders goals and design decisions will be used [18, 24]. In the ArchiMate project [14, 30], some initial work into the capturing of architectural design decisions was conducted as well. This preliminary work provides a good starting point to make architecture-level design decisions more explicit. The motivation extension of the new 2.0 version of the ArchiMate standard [10] further extends this work.

## 2.4 Logical formalisms for decision making

One of the goals of this research is to apply formalisms from domains such as computer science to enterprise architecture. To do so, we focused our attention on formalism coming from logics and artificial intelligence, used to reason on decision making process. To this point, we focused on decisions made in presence of exterior beliefs and reconsideration of these decisions. We also believe that agreement technologies, and in particular normative systems, are an interesting track for future investigations.

**Planning and intention revision:** In terms of logics, design decisions are treated as plans, in which the assumptions will be represented in a variety of ways, depending on the nature of the assumptions. Techniques for reasoning about assumptions in plans are adopted from theories of diagnosis, truth maintenance systems [2, 3, 26], logic programming, non-monotonic logic, and most recently from assumption based formal argumentation [4]. Classical planning techniques are often not sufficient, therefore they have been extended with the theory of intentions [11, 22, 31, 25]. Shoham [11, 27] recently changed the focus on intentions from its historical philosophical perspective to a computer science database perspective of revising plans in the context of beliefs.

**Agreement technologies:** Due the rise of distributed systems over the years, the focus shifted from individual to collective reasoning, with formal theories of multi-agent systems reasoning about interaction among multiple actors or stakeholders [1, 15]. Members of the COST Action ICO801-AT<sup>4</sup> have divided the action in 5 working groups/topics (Semantics, Norms, Organisations, Argumentation and Negotiation, Trust). Our focus was on the normative issues. The issuing of norms has the characteristic of stating norms in such a form in which allows them to regulate a wide range of situations, to be stable over a long period of time. Grossi [8] makes a distinction between abstract norms and concrete/refined norms. We can make a clear connection between normative refinements and architecture principles: the principles intervene in the non-functional level, the strategic level, are derived from drivers (goals of stakeholders associated with the current situation), and describe qualitatively the architecture. Requirements are more specific and provide functional guidelines.

## 3 Research methodology

Methods for developing design theories must specifically account for the aspects of rigour and relevance that according to Hevner *et al.* [9] shape the environment in which design science research takes place. This project will follow the design research process as outlined by Peffers *et al.* [23].

Advanced techniques from knowledge representation and multi-agent systems, as presented in section 2, will be applied and generalized to model the dynamics and uncertainty characteristic of such a socio-epistemic context as enterprise architecture. In order to do so, the project will have 3 iterations,

---

<sup>4</sup> <http://www.agreement-technologies.eu>

as follows: [1] Capture design rationale and basic reasoning: develop a formal framework in order to capture enterprise architecture decisions. The validation of the framework will be done using with case studies, conducted in collaboration with practitioners and companies. [2] Reasoning on persistence of assumptions: introduce uncertainties, refine the framework, apply more advance reasoning mechanisms (e.g Bayesian networks and causal theories). [3] Select and develop relevant reasoning mechanisms that allow for multi-stakeholder situations and associated negotiations (e.g apply dynamic game theoretic based approaches, description logics)

## 4 Preliminary results

The preliminary results, regarding representations of decisions, principles and underlying assumptions, as well as two decision reconsideration algorithms, were published in [16, 17].

A paper linking IT security approaches (attack defence trees [12, 19]) with risk management and goal oriented design was recently accepted [28]. The aim of this paper is to provide a framework that enables stakeholders to achieve their goals by performing risk analysis.

## 5 Conclusions and future work

In the previous sections we have identified a research problem, we formulated the adequate research questions, position our work and presented preliminary results.

In the short term, we have two important directions to follow. First, to get in touch with industrial partners and get involved in a company's case study. Second, we intend to develop further the preliminary results regarding risk management based architecture principles and validate our results within a company.

In the longer term, the project will result in a logic-based framework to capture the rationalization of architecture related design decisions, and to reason about the relationship between these decisions and their underlying assumptions. We follow an iterative approach for development and validation, as presented in section 3. The framework will cater for uncertainties of the underlying assumptions, as well as negotiation between different stakeholders involved in the creation and implementation of architectures. The framework will be specialized further towards two classes of properties of enterprises and their IT: security and modifiability. In the meantime, the relevance of the results will have been validated in terms of a number of real-world case studies.

## Acknowledgements

The author wishes to thank her daily co-supervisor, Dr. Khaled Gaaloul, for many hours of discussions and feedback regarding the research topic and for

his help in planning the course of the research. Furthermore, she would like to thank her supervisor, Prof. Dr. Erik Proper, and her former thesis coordinator, Prof. Dr. Leon van der Torre.

## References

1. H. Billhardt, R. Centeno, C. E. Cuesta, A. Fernández, R. Hermoso, R. Ortiz, S. Ossowski, J. S. Pérez-Sotelo, and M. Vasirani. Organisational structures in next-generation distributed systems: Towards a technology of agreement. *Multiagent and Grid Systems*, 7(2-3):109–125, 2011.
2. J. de Kleer. A perspective on assumption-based truth maintenance. *Artif. Intell.*, 59(1-2):63–67, 1993.
3. J. Doyle. Reason maintenance and belief revision - foundations vs. coherence theories. In *Belief Revision*, pages 29–51. Cambridge University Press, 1992.
4. P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artif. Intell.*, 77(2):321–358, 1995.
5. W. Engelsman, H. Jonkers, and D. Quartel. ArchiMate Extension for Modeling and Managing Motivation, Principles and Requirements in TOGAF. White paper, The Open Group, August 2010.
6. D. Greefhorst and H. Proper. *Architecture Principles – The Cornerstones of Enterprise Architecture*. Enterprise Engineering Series. Springer, Berlin, Germany, 2011.
7. D. Greefhorst and H. Proper. *Architecture Principles – The Cornerstones of Enterprise Architecture*. Enterprise Engineering Series. Springer, Berlin, Germany, 2011.
8. D. Grossi. *Designing Invisible Handcuffs - Formal Investigations in Institutions and Organizations for Multi-Agent Systems*. PhD thesis, University of Utrecht, 2007.
9. A. Hevner, S. March, J. Park, and S. Ram. Design Science in Information Systems Research. *MIS Quarterly*, 28:75–106, 2004.
10. M.-E. Iacob, H. Jonkers, M. Lankhorst, and H. Proper. *ArchiMate 2.0 Specification*. The Open Group, 2012.
11. T. F. Icard, E. Pacuit, and Y. Shoham. Joint revision of beliefs and intention. In *KR*, 2010.
12. B. Kordy, S. Mauw, M. Melissen, and P. Schweitzer. Attack-defense trees and two-player binary zero-sum extensive form games are equivalent. In *Proceedings of the First international conference on Decision and game theory for security, GameSec’10*, pages 245–256, Berlin, Heidelberg, 2010. Springer-Verlag.
13. M. Lankhorst et al. *Enterprise Architecture at Work: Modelling, Communication and Analysis*. Springer, Berlin, Germany, 2005.
14. M. M. Lankhorst. *Enterprise Architecture at Work - Modelling, Communication and Analysis (4. ed.)*. The Enterprise Engineering Series. Springer, 2013.
15. T. Malone and K. Crowston. The interdisciplinary study of coordination. *Computing Surveys*, 26 (1):87–119, 1994.
16. D. Marosin, H. A. Proper, and L. van der Torre. Changing Agreements: Intention Reconsideration based on Assumptions and Reasons. In S. Ossowski, F. Toni, and G. A. Vouros, editors, *AT*, volume 918 of *CEUR Workshop Proceedings*, pages 262–263. CEUR-WS.org, 2012.



17. D. Marosin and L. van der Torre. Changing commitments based on reasons and assumptions. In *the 12th AAMAS conference, To Appear in COIN workshop*, 2013.
18. R. Mason and I. Mitroff. *Challenging Strategic Planning Assumptions*. John Wiley & Sons Ltd, 1982.
19. S. Mauw and M. Oostdijk. Foundations of attack trees. In *International Conference on Information Security and Cryptology ICISC 2005. LNCS 3935*, pages 186–198. Springer, 2005.
20. M. Op 't Land, H. Proper, M. Waage, J. Cloo, and C. Steghuis. *Enterprise Architecture – Creating Value by Informed Governance*. Springer, Berlin, Germany, 2008.
21. A. Osterwalder and Y. Pigneur. *Business Model Generation: A Handbook for Visionaries, Game Changers, and Challengers*. Self Published, Amsterdam, The Netherlands, 2009.
22. S. Parsons, O. Pettersson, A. Saffiotti, and M. Wooldridge. Intention reconsideration in theory and practice. In *ECAI*, pages 378–382, 2000.
23. K. Peffers, T. Tuunanen, M. Rothenberger, and S. Chatterjee. A design science research methodology for information systems research. *Journal of Management Information Systems*, 24(3):45–77, 2007.
24. C. Potts and G. Bruns. Recording the reasons for design decisions. In *Proceedings of the 10th International Conference on Software Engineering*, pages 418–427, 1988.
25. A. S. Rao and M. P. Georgeff. Intentions and rational commitment. In *Proceedings of the First Pacific Rim Conference on Artificial Intelligence (PRICAI-90)*, 1993.
26. R. Reiter and J. de Kleer. Foundations of assumption-based truth maintenance systems: Preliminary report. In *AAAI*, pages 183–189, 1987.
27. Y. Shoham. Logical theories of intention and the database perspective. *Journal of Philosophical Logic*, 38:633–647, 2009.
28. S. Sousa, D. Marosin, K. Gaaloul, and N. Mayer. Assessing Risks and Opportunities in Enterprise Architecture using an extended ADT approach. In *the 17th IEEE International EDOC conference*, To Appear 2013.
29. The Open Group. *TOGAF Version 9*. Van Haren Publishing, Zaltbommel, The Netherlands, 2009.
30. G. Veldhuijzen van Zanten, S. Hoppenbrouwers, and H. Proper. System Development as a Rational Communicative Process. *Journal of Systemics, Cybernetics and Informatics*, 2(4):47–51, 2004.
31. M. Wooldridge and S. Parsons. Intention reconsideration reconsidered. In *ATAL*, pages 63–79, 1998.