

Introducing Argumentative and Discursive Enterprise Leading and Management

Sebastian Bittmann¹, Balbir Barn², Tony Clark², Oliver Thomas¹

¹ University Osnabrueck, Germany

² Middlesex University, London

Abstract. Leading an enterprise requires, obviously, decision making. However, these decisions require explanations in order to make it possible for stakeholders to get an understanding about the enterprise's strategic direction. This is even more important when these stakeholders are in charge to transpose such strategic decision into their tactical or operational work. Enterprise modelling may be capable of depicting strategies per se, but it is rather a vessel of communication than of explanation. Whilst, a strategy may be accordingly modelled, those who receive such a model needs to purposeful interpret and successfully implement it. However, without any insights, justifications or references that go beyond the claim of a model, it is difficult to embrace the theory of the actual modeller. Therefore, in this paper argumentative modelling will be specifically applied to the domain of strategic management. Moreover it will be elucidated how modelled strategic arguments can be used as a basis for enterprise architecture alignment and management. As it will be shown in the paper, the application of argumentative modelling overcomes classical restrictions and makes it possible to support a discourse, which can be later on used as an explanation for the intentions of the modeller.

1 Introduction

Strategies are a central component of the enterprise's success [1], as they guide the enterprise within an often unstable environment as well as they position the enterprise within a competitive and challenging market. So, whether the strategies are planned or emergent [2], they decide about the transition of the external requirements into business values by means of the enterprise's resources and capabilities. Although the strategy of an enterprise shall be incorporate by every artefact of the enterprise, the impact is often difficult to measure. This relates mainly to the often natural-language-based spread strategies [3]. Despite the available approaches for providing a strategy by means of a conceptual model [4] and further the consideration of strategic aspects in enterprise architectures [5], a possible benefit of the conceptualisation of a strategy is missing. For example, based on the strategy itself, the *management*, the *argumentative evaluation* of and *reasoning* about IT-artefacts is not possible, because of rather straightforward perspective on enterprise transformations [6, pp. 11-14]. Therefore the

motivation of conceptualising the strategy for documentary reasons may be not sufficient to take the extra effort.

In order to motivate the relevance for conceptualising a strategy accordingly, in this paper, the concept of a strategic argument will be introduced that was derived from the theory of argument by STEPHEN TOULMIN and explicitly from the conception of such an argumentative theory for modelling languages, namely the Argumentative Modelling Language [7]. Thereby, a relation between an IT-artefact and the respective part of the strategy becomes obvious, which enables a possible justification of the IT-artefact and further the motivation for an adaptation of the artefact based on changing conditions. The rest of the paper is outlined as follows. Initially, the theoretical conception behind the strategic argument is introduced in section 2. Following, the conception of the strategic argument as well as its relation towards both, the strategic and enterprise architecture management will be discussed in section 3. Successively, the approach will be evaluated in a case study in section 4. Lastly, the paper ends with a conclusion in section 5.

2 The Argumentation Modelling Language

The Argumentation Modelling Language (ArgML) was derived from the work by STEPHEN TOULMIN, who proposed an argumentation theory from the field of jurisprudence [8]. Using a form of argument as proposed by the ArgML, enables the depiction of the theory that lies behind a conceptual model [9, 10]. Moreover the form of argument as proposed by TOULMIN enables an evaluation of the underlying theory [11]. The conception of the ArgML is a complete exclusion of natural language and exclusive to semi-formal modelling languages, respectively domain-specific modelling languages. So any argument proposed by means of the ArgML follows a strictly specified syntax and is interpretable by clearly defined semantics. Therewith the abstract syntax of the ArgML is given by Figure 1. A detailed explanation of the key concepts that are either adopted based on TOULMIN'S theory or added with respect to requirements of the formalisation process.

The prime concept is the *argument*. Any argument is a container for a multitude of claims and their rebuttals. An arguments comprises claims that are specified by means of an uniquely chosen *language*. Generally, such a language should satisfy the purposes of the resulting model, so usually a domain-specific language should be chosen that can be characterised as semi-formal. Therewith, its specification offers various concepts for expressing claims. Every *claim* that is included by the argument and specified by the respective language embodies knowledge, which shall be established. It is expressed through a model, which follows the syntax of the respective argument's associated language. Respectively, a model is an instance of the chosen language of an argument used for expressing the designated claims. As the knowledge offered by claims may become established, those claims will be delegated by *grounds* in order to propose, respectively establish, new and upcoming knowledge by means of new claims. To

previous analyses. After the specification, the strategy will be *implemented and enacted*. Hence, the strategy finds its concretisation by means of the different actions taken by the employees of the enterprise. Finally, the proper implementation needs to be audited during the *control* phase. Mainly the two initial phases are of relevance for the proposed approach, as these focus rather on concepts. With respect to the argumentation theory, the analysis phase provides facts on which strategic steps that claim the complete strategy may be grounded. Additionally, further concepts are needed that justify a strategic step on the facts of the analyses.

3.2 Strategic Arguments

Applying the ArgML to a specific domain, namely the strategic management, requires the various concepts of the ArgML to become domain-specific. Respectively, the concepts need to be adapted or specialised towards the specific requirements of the specific domain. Thereby, the ArgML represents a certain reference model [13], which has to be adapted towards the requirements of the specific domain, namely strategic management. Therefore, the ArgML has to support the grounding of claimed strategic steps. Based on their grounding, it needs to be possible to justify the strategic steps regarding their expected benefit and practicability [14, pp. 53-130].

So arguing a strategy requires at least two parts: the explanation of the *possibility*, respectively if it's possible to resolve the strategy and there is a need for explaining the *expected benefit* is. Both these parts can be structured by means of the concepts of the ArgML. Beginning with the *claim*, which represent a certain statements that is sought to be established; its purpose is the proposition of a specific, rather atomic, strategic step. Based on this multitude of claims, those that ultimately become valid form the actual strategy of the enterprise [4].

Nevertheless, the proposed claims need to be grounded in order to justify the strategic argument with respect to the expected benefit and the practicability. Thereby, a *ground* of the ArgML is specialised by a strategic argument to any circumstance either owned by the enterprise as a form of resource or inhabited in the external environment. Accordingly, grounding on the enterprise's resources further justifies the possibility to implement the strategic step by the enterprise. Grounding on the environment, enables the elucidation whether a strategic step leads to a better position within the market or improves the overall competitiveness. So, a ground is a statement about an irrefutable state of the enterprise or its environment. So with having both concepts applied to the strategic management domain, the phase of analysis and specification of the strategic management can be purposefully supported by means of conceptual models.

To simply refer to a ground from a claim jeopardises the chance for insights, rebuttals and onward improvement [15]. Thereby, explanations are required that describe a certain design for realising a specific set strategy, against other possible design decisions. The concepts of a *qualifier*, enables a more specific statement about the required resources and external requirements. The qualifier embodies a certain set of rules that are capable of evaluating the grounds properly in order

to make a statement about the practicability of the strategic step. If for example, the enterprise owns a certain IT-System, a strategic argument might refer to such a resource. However, the qualifier is able to state the exact amount of time this IT-System will be needed for the realisation. Thereby, refutations between two arguments, namely *rebuttals* can be identified based on the references and whether the respective requirements can be aligned or not. The concept of a *warrant* in a strategic argument represent the possible benefit the enterprise might have with following a specific strategic step. Hence, it must provide the conclusion for realising a strategic step, which is in the case of strategic management, the achievement of a specific strategic goal. Such a goal represents a change in the external or internal circumstances that ultimately should lead to a better position of the enterprise. Thereby within the perspective of argumentation, a strategic goal shall be viewed as a transition from a current to a desired future state. Next to the desired benefits, the warrant further has to provide the expected effort, which has to further be accounted by the prediction of a future state. On account of this, the warrant reveals the sense of pursuing a strategic step based on the circumstances of the enterprise. Ultimately the general conceptualisation can be gained from Figure 2.

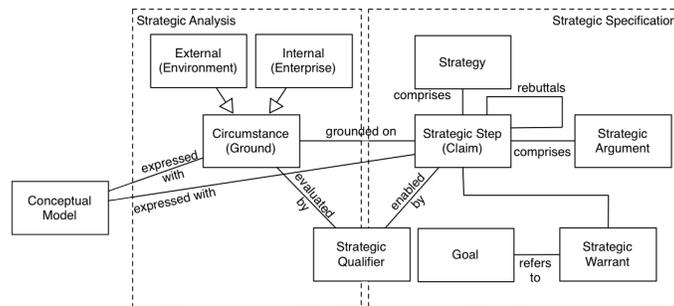


Fig. 2. Conception of a Strategic Argument

3.3 Implementation of Strategic Arguments within an Enterprise Architecture

As it was elucidated in the previous chapter, the strategic arguments can be used for the specific requirements evolved with reference to the analysis and specification of the enterprise’s strategy. However, in order to illustrate the beneficial contribution, the concept of enterprise architecture management as a reference for the evaluation of possible applications of strategic arguments within the enterprise. An enterprise architecture subsumes the relevant artefact for the enterprise and their interrelation [16, 17]. In general, these artefacts are considered with strategical, such as product and services, as well as process, application and technological related issues [5, 18].

For the general purposes of this paper, the LEAP approach [19] was chosen, which represents a lightweight, component and layer based approach for enterprise architecture management. The general conception of a component as an artefact enables the illustration of general rules for applying the strategic argument throughout the enterprise architecture. Furthermore, the layered perspective of LEAP, fosters the refinement of strategic arguments to concrete software components and the generation of strategic directions based on technological innovations. Hence, the benefits of the approach can be elucidated by the use of a middle-out approach for enterprise arguments, without restricting the approach to the strategy and rather top-down or the technology and rather bottom-up, but enabling a possible alignment between technological and organisational innovations [20]. LEAP, although supports graphical representation by means of diagrams, uses textual representations. Respectively LEAP uses OCL statements for definitive specification of the enterprise architecture. The most general statement is " $(C,o) [n=v; \dots]$ when Q ", whereby C is a class name, o an object identifier, n a name of a specific field and v is a value. Q is an OCL constraint that should satisfy the proper creation of o . So in order to enable the use of strategic arguments in LEAP, the form of statement needs to be properly adapted. Therefore it is necessary to enable references to enable a justification of the decisions. Respectively, whilst it is already possible to propose claims, it must be further necessary to ground these claims. So with

$(C,o) [n=v; \dots]$ requires (R, QS) targets (G, WS) when Q

a specific form of the OCL constraint is given that satisfies the requirements of the conception of a strategic argument. Therefore R has been introduced for referencing an already available artefact of the enterprise architecture or an external circumstance and QS describes how this reference qualifies as a ground for the argument. Additionally G references any goal that should be targeted by means of the introduction of the artefact o and WS gives the justification of the achievement of the respective goal by means of the introduced artefact.

4 Case Study

In the following, an evaluation of the respective introduced approach will be undertaken by means of a case study. The case study focuses on an enterprise, whose primary business is the online retail of commercials. For that particular purpose, the company uses an own developed platform, on which the respective customers can purchase, request customer services and other services regarding compliance. Thereby, the enterprise mainly focuses on extending its customer base and additionally the increase of the customer value by means of cross- and up-selling.

The upcoming Figure 3 illustrates the enterprise architecture on a current state and additionally, several arguments that claim adaptations within the enterprise architecture. So, in an excerpt, the enterprise architecture includes three different business processes that were proposed to target the achievement of the

initial strategy. These business processes are further supported by certain applications that were coordinated on the application layer. Ultimately, the enterprise architectures include certain technology for the realisation of the application systems. However, while the workflow management system has already been introduced, it hasn't been yet used for the realisation of an application system or a business process.

So, initially, as depicted by Figure 3, an automation of the customer services was proposed based on the respective goal, namely the sale increase. This was grounded on the opportunities a workflow management system offers and additionally, the available time of the customer consultants that prior were in charge for executing the services. So, based on the ground and the respective goal to achieve and strategy to resolve, the argument proposes an automation or partly automation of the business processes "Order Processing" and "Customer Services". The formalisation of this argument regarding customer services is given in the following.

```
context CustomerServices
self.includeComponent(CRM[components=(WfMS, CustomerDatabase), ...])
and self.includeComponent(SelfServicePortal[components=WfMS, ...])
  requires CustomerConsultants.avgWeeklySpentTime(OrderProcessing)
    >36000s and TechnologyLayer.includes(WfMS)
  targets SaleIncrease >= 0.1
```

Additionally, a further argument proposed the focus on the customer relation with the motivation of an increase of five per cent of the customer value. Such argument is grounded on the customer requirements and additionally, on the previous analysed potential of cross- and up-selling. So, in order to resolve such a strategy, the business processes of customer services and customer compliance were sought to adapt to the upcoming and *justified* customer needs, as given below.

```
context CustomerServices
self.includeComponent(CustomerServiceSystem
  [components=(CustomerServices, CustomerDatabase), ...])
and self.excludeComponent(WfMS)
  requires CustomerReBuyPotential>0.5
  targets AvgRevenuePerCustomerIncrease>0.05
```

However, on a later stage, respectively the design of the respective business processes based on the proposed strategic arguments, it occurs that although prior in harmonisation, two of the goals provided by the strategic arguments are in conflict. Such a conflict results from the necessity for the process "Customer Services" for being compliant to both prior discussed strategic arguments. With the current business process landscape of the company, the respective business process has to be designed with reference to automation potential as well as customer relation. These directives are in conflict, as the customer relation needs to be fostered by a customer consultant and requires human interactions. Hence, the identification of a rebuttal within the set goals requires a specific level of concretisation by the respective strategic step.

In parallel and untouched from this conflict, a request from the operative has arisen that demands the integration with the ticket manager and the customer

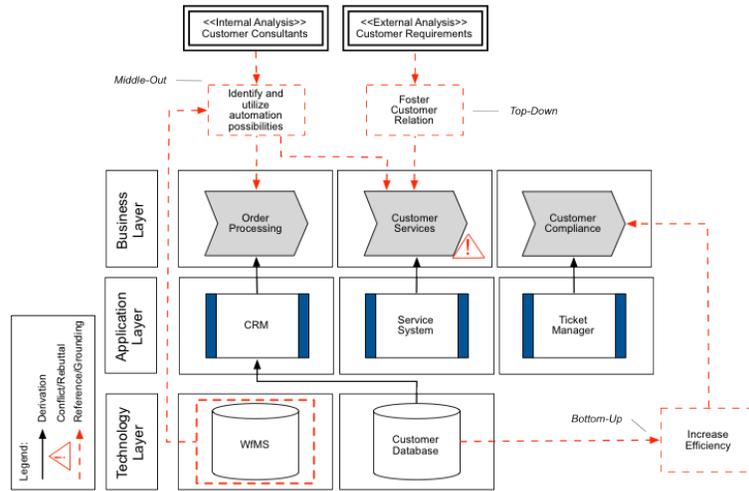


Fig. 3. Initial Derived Enterprise Architecture on a Conceptual Layer supplemented with Strategic Arguments

database, which was initially only used by the CRM System, for the more efficient performance of the customer compliance process.

```

context CustomerCompliance
self.includeComponent(TicketManager.includeComponent(CustomerDatabase))
  requires TicketManager.datamodel
    .intersection(CustomerDatabase.datamodel).includes(Customer)
  targets EfficiencyIncrease > 0.1

```

So, as depicted by Figure 3, the actual conflict between two arguments only become visible, after the conceptual realisation of the respective artefacts, as the design requires balancing between opportunities. Thereby, design decisions within the business processes are contrary to each other, but both were based on a strategic direction that initially didn't reveal a conflict.

However, with the identification of the actual conflict and documentation of the design decisions, the different strategic steps can be purposefully refined in order to propose a less ambiguous strategy with respect to enterprise architecture management. With the strict grounding on the specific *targets* and *requirements* of the artefact, the alignment to the business strategy of the company was possible, as inference could be made to the initial intentions, which was derived from the overall enterprise strategy. Moreover with the refinement of the strategic arguments, the enterprise architecture can be purposefully aligned. With a more elaborated strategy as well as the introduction of additional artefacts, the alignment was feasible, without jeopardising the overall strategy implementation. The upcoming Figure 4 represents the aligned enterprise architecture that specifically embodies the adapted artefacts towards the three initially proposed strategic arguments. Specifically, the strategic steps that aimed at the customer

relation were adapted. Therewith a self service business process was added with reference to those services that won't necessarily benefit from human interactions.

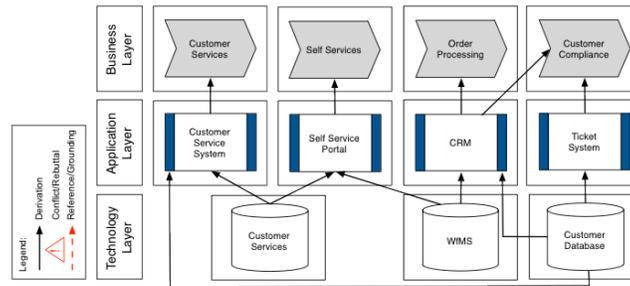


Fig. 4. Aligned Enterprise Architecture based on Strategic Arguments on a Conceptual Layer

5 Conclusion

With the presented approach, a manner of enterprise architecture management was introduced that uses a form of argument for purposefully managing the evolution of the enterprise architecture in order to enable the purposeful alignment with the overall enterprise strategy. With the use of arguments a misalignment between the strategy and the enterprise architecture becomes identifiable with a concretisation of the actual artefacts by means of their design directives. Thereby, design decisions of artefacts can be supported by their underlying rationale derived from the strategic directives. As on first sight, strategic steps seem in harmony, later on, a misalignment can be revealed based on a more complete specification. With having the design decision's rationale the adaptation as well as the alignment of the respective artefacts can become more directed and contributing. Additionally, the approach supports bottom-up and middle-out, next to rather strategic top-down, implementations and thereby, is able to return developed insights to the overall strategy of the enterprise.

References

1. Porter, M.E.: Competitive Strategy: Techniques for Analyzing Industries and Competitors. The Free Press, New York (1980)
2. Mintzberg, H., Waters, J.A.: Of strategies, deliberate and emergent. Strategic Management Journal **6**(3) (July 1985) 257–272
3. Hoppenbrouwers, S., Proper, H.A., Van Der Weide, T.P.: A Fundamental View on the Process of Conceptual Modeling. In Delcambre, L., Kop, C., Mayr, H.C.,

- Mylopoulos, J., Pastor, O., eds.: Proceedings of the 24th International Conference on Conceptual Modeling. Volume 3716 of Lecture Notes in Computer Science. Springer, Berlin, Heidelberg (2005) 128–143
4. Kaplan, R.S., Norton, D.P.: Having trouble with your strategy?: Then map it. In: Focusing Your Organization on Strategy - with the Balanced Scorecard. 2nd edn. Harvard Business School Publishing Corporation (2000)
 5. Winter, R., Fischer, R.: Essential layers, artifacts, and dependencies of enterprise architecture. *Journal of Enterprise Architecture* **3**(2) (2007) 7–18
 6. Greefhorst, D., Proper, E.: Architecture Principles: The Cornerstones of Enterprise Architecture(Google eBook). (2011)
 7. Bittmann, S., Barn, B., Clark, T.: A Language Oriented Extension to Toulmins Argumentation Model for Conceptual Modelling. In: 22nd International Conference on Information Systems Development (ISD2013). (2013)
 8. Toulmin, S.E.: The Uses of Argument. Updated edn. Cambridge University Press, Cambridge, UK (2003)
 9. Barn, B.S., Clark, T.: A domain specific language for contextual design. In Bernhaupt, R., Forbrig, P., Gulliksen, J., Lárusdóttir, M., eds.: Human-Centred Software Engineering. Volume 6409. Springer, Berlin Heidelberg (October 2010) 46–61
 10. Barn, B.S., Clark, T.: Revisiting Naur’s programming as theory building for enterprise architecture modelling. In: CAiSE’11 Proceedings of the 23rd international conference on Advanced information systems engineering, Berlin, Heidelberg, Springer (June 2011) 229–236
 11. Gregor, S.: The nature of theory in information systems. *MIS Quarterly* **30**(3) (September 2006) 611–642
 12. Näsi, J.: Information systems and strategy design. *Decision Support Systems* **26**(2) (August 1999) 137–149
 13. Thomas, O.: Management von Referenzmodellen: Entwurf und Realisierung eines Informationssystems zur Entwicklung und Anwendung von Referenzmodellen. Logos, Berlin (2006)
 14. Johnson, G., Scholes, K., Whittington, R.: Exploring Corporate Strategy: Text and Cases. (2008)
 15. Plataniotis, G., Kinderen, S., Proper, H.: Capturing Decision Making Strategies in Enterprise Architecture A Viewpoint. In Nurcan, S., Proper, H., Soffer, P., Krogstie, J., Schmidt, R., Halpin, T., Bider, I., eds.: Enterprise, Business-Process and Information Systems Modeling SE - 24. Volume 147 of Lecture Notes in Business Information Processing. Springer Berlin Heidelberg (2013) 339–353
 16. Zachman, J.A.: A framework for information systems architecture (1987)
 17. Aier, S., Riege, C., Winter, R.: Unternehmensarchitektur Literaturüberblick und Stand der Praxis. *WIRTSCHAFTSINFORMATIK* **50**(4) (September 2008) 292–304
 18. Rychkova, Irina; Wegmann, A.: A Method of Functional Alignment Verification in Hierarchical Enterprise Models. In Latour, T., Petit, M., eds.: Proceedings of the CAISE06 Workshops and Doctoral Consortium, Namur, Presses universitaires de Namur (2006) 244–253
 19. Clark, T., Barn, B.S., Oussena, S.: LEAP. In: Proceedings of the 4th India Software Engineering Conference on - ISEC ’11, New York, USA, ACM Press (February 2011) 85–94
 20. Henderson, J.C., Venkatraman, N.: Strategic alignment: leveraging information technology for transforming organizations. *IBM Systems Journal* **32**(1) (January 1993) 4–16