Enterprise Governance and DEMO

Guiding enterprise design and operation by addressing DEMO's competence, authority and responsibility notions

Miguel Henriques, José Tribolet, and Jan Hoogervorst

Department of Information Systems, Instituto Superior Técnico (IST-UTL), Lisboa
miguel.henriques@ist.utl.pt,jose.tribolet@inesc.pt,jan.hoogervorst@sogeti.nl

Abstract. The lack of an organizational competence that embodies the capacity to restrict the enterprise undesirable design freedom and guide the subsequent operation, from a holistic point of view, leads to incoherence and inconsistency among the enterprise elements. The research brings forward the importance of this organizational competence labeled enterprise governance (EG) in defining DEMO's ontological models and using their subsequent authority, responsibility and competence notions to guide the enterprise dynamics. Based on these results, the article provides a reference method for the EG to define a set of normative outputs, derived from these three notions addressed in the enterprise ontologic models, comprising a set of principles to address enterprise integration and a set of rules to deal with on-going organizational changes while addressing security issues.

Key words: Enterprise Governance, Enterprise Design, Enterprise Architecture, Enterprise Ontology, Enterprise Engineering

1 Introduction

In a world of growing business dynamics, high rates of technological and organizational changes, enterprises need to be continuously (re)designed and (re)engineered in order to achieve strategic and operational success. Our research will be built around the enterprise development theory within the enterprise engineering discipline.

In this context, we address one core problem: the lack of coherence and consistency among the various enterprise elements resulting from the enterprise incapacity to effectively build the enterprise strategy into design and manage the subsequent changes at the operational plane from a holistic point of view [11, 1]. It is estimated that between 70% and 90% of strategic initiatives tend to fail [10, 3]. Researchers argue that such failures in most cases result from inadequate strategy implementation in the sense that if the enterprise aspects are not addressed in design by thinking about the enterprise as an organic whole the enterprise will not be able to operate as a unified system and the strategy implementation will tend to fail [1].

We argue that there are two interrelated main reasons behind this problem. First, the enterprise does not have the ability to apply in practice the design theory from the enterprise engineering discipline and thus, it is unable to master the enterprise complexity and to develop an integrated enterprise system [3]. Second, absence of an organizational competence that should guide globally the enterprise development process and subsequent changes in order to ensure the correct use of this theory [7].

INForum 2010 - II Simpósio de Informática, Luís S. Barbosa, Miguel P. Correia (eds), 9-10 Setembro, 2010, pp. 473–476

2 An organismic governance approach

The systemic approach to problems focuses on systems taken as a whole, instead of their parts taken individually [1, 11]. Within this view an enterprise is perceived as a goal-seeking system, intentionally designed by a set of interacting human beings behaving according to assigned authority and corresponding responsibility against a common background of social norms and values. Mastering the enterprise complexity and guide the design[1] in a holistic manner is essential during the implementation of the strategy to achieve integration among enterprise components [13]. However, this notion of guidance is often associated with a 'mechanistic' perspective (top-down, management and control oriented). Within this perspective, there is no adequate competence and attention for addressing the enterprise design (we refer to [9, 11]).

Enterprise strategy and design subjects transcend the capabilities of the corporate governance and IT governance disciplines and can only be addressed within the overall scope of EG (we refer to [6]). The EG consists in an integrated whole of knowledge, skills and technology, whereby employees are viewed as the crucial core for continuously exercising guiding authority over enterprise strategy and *architecture* development, and the subsequent design, implementation and operation.

Architecture notion has been associated to a descriptive approach perceived as a "blueprint" of the system construction and a prescriptive approach concerning design guidance [5, 8, 12]. A prescriptive approach of the concept must be exercised comprising "consistent and coherent set of design principles" defined by the EG in order to guide the design by restricting its undesirable freedom [3].

3 Ontology - DEMO methodology

The models resulting from the design process approach the system construction at different levels of abstraction. At the "highest level" there is the ontological model and at the "lowest level" there is the implementation model [3]. The core meaning of system ontology in our thesis context is a model for describing and understanding the construction and operation fully independent of the way the system is implemented which is coherent, comprehensive, consistent and concise (we refer the reader to [2]).

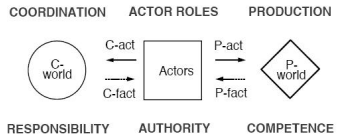

Fig. 1: The construction axiom[2]

Competence	"The ability of a subject to perform a particular type of P-act as well as the corresponding C-acts."
Authority	"The condition of being allowed to act." It can be assigned through authorization and delegation.
Responsibility	"The quality of a subject to be committed to the coordination facts he or she has performed, as well as to coordination acts that are addressed to him or her."

Fig. 2: Authority, responsibility and competence[2]

The scientific root of DEMO is the Ψ-theory, we outline its four essential axioms according to [2]. *The construction axiom* indicates that an enterprise consists of actors performing productions acts (P-acts) to bring about the enterprise mission and coordination acts (C-acts) to enter into and to comply with commitments. *The operation axiom* says that for every type of C-act there is an action rule to guide enterprise actors. *The transaction axiom* argues that P-acts and C-acts occur in generic recurrent patterns performed by two actors called transactions. *The abstraction axiom* distinguishes three human abilities to perform C-acts: forma, informa and performa.

[1] Design is percieved as the production process of conceptual models of a system [3].

4 Proposed model and underlying reference method

The conceptual model that relates the notions of EA, ontological models and EG is outlined in figure 3. The EG uses the ontological models to master in a holistic manner the enterprise complexity and devise a set of outputs that will guide the enterprise design as well as the enterprise execution plan (operation). Governance outputs can be divided in (1) principles devised from all the enterprise design domains (traceable with the enterprise areas of concern) that will restrict the design process (EA notion), and (2) outputs retrieved from DEMO models based on the notions of competence, authority and responsibility, which will guide the detailed design and the enterprise operation. Principles purpose is to deal with the article main concern (the lack of integration among the enterprise elements at design level), while rules will ensure that the enterprise operation conforms the enterprise design.

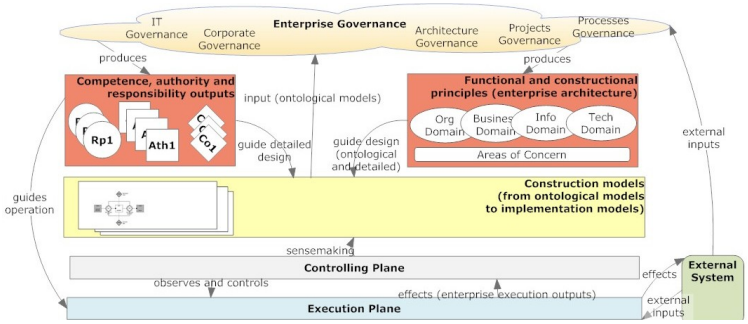

Fig. 3: Enterprise Governance, Enterprise Ontology and Enterprise Architecture

Based on DEMO's theory and governance themes discussed above, as well as researches in the field of responsibility and security modeling [4], we infer a reference method to govern the enterprise dynamics with DEMO depicted in table 1.

Method stage	Observations
(1) identify the enterprise actor roles	The task of identifying actor roles is already provided by DEMO: they can be retrieved from the enterprise construction (interaction or interstriction) models.
(2) identify the areas of responsibility	For one specific transaction, the Process Model defines all the C-acts that an actor role is allowed to perform. Hence, the responsibility areas are rigorously defined in DEMO's Process Models
(3) identify the competence domains and define a set of competence principles for each actor role	Competence domains can be perceived as attributes that will guide the evaluation process to check if a person has the adequate competence to exercise its job. Competence principle purposes to restrict the detailed design freedom regarding the actors production acts.
(4) define all the authority rules for each actor role (who has the right to exercise authorization and delegation and in which conditions)	For this purpose should be defined (1) the acts that each actor need to do, (2) who is allowed to access what information and the information that must be audited for each actor role. The illustration of these three requirements can be further transformed in eligible authorization rules. Consequently, if the acts and information required do not exist, the actors are not allowed to do and see anything else than what is specified. In this fashion, DEMO enforces a role-based access control.
(5) define the action/coordination rules for each type of C-/P-act	operational rules consist of two categories: the coordination rules for guiding the coordination activities (responsibility) and the production rules for guiding production activities (competence).
(6) identify responsibilities pre-conditions and post-conditions needed to discharge a responsibility	Since the c-acts are represented as action rules, then we can assume that there is a set of pre-conditions that must hold before an action rule can start. After an instance of discharging a responsibility there are statements about the environment and the agent that are true, these are the post-conditions (formal statements).
(7) create a list of exception conditions	Exception conditions list express all the exceptions that need to be handled when occurs a deviation regarding the enterprise norms. In this fashion, when an exception is detected the adequate mechanisms and actors will be properly alerted

Table 1: Proposed reference method

The reference model and method are being validated in the Portuguese Justice System, in particular at DIAP. The benefits of these artifacts are being demonstrated as essential to identify and deal with the inconsistent and incoherent requirements by devising principles from a global architecture framework to support the operation of all the justice procedures and actors (e.g. the lack of coordination between internal and external entities in crime investigations), and to deal with issues such as security, information access, traceability of the agents' actions, among others, which are addressed at the design level and its correct execution ensured at the operational level in order to deal with the continuous organizational changes.

5 Conclusion

This article described the potential of bringing together the notions of enterprise governance and enterprise ontology (within DEMO). On one hand, the EG should be associated to an organismic perspective responsible for guiding the enterprise strategy and enterprise development by restricting the undesirable design freedom in the form of principles (architecture notion) and guiding the subsequent enterprise operation in the form of operational rules. On the other hand, DEMO provides a methodology to represent the enterprise essence in an intellectual manageable way.

Based on the research in this field of knowledge, we developed a conceptual model and an underlying method to support the EG in defining a set of normative outputs. This method uses the notions of competence, responsibility and authority within DEMO to deal with the continuous changes and restrict the detailed design process (competence principles regarding P-acts), to deal with security issues associated to information access and responsibility transfer (authority rules), and to help identifying requirements that are inconsistent and incoherent and mutually align enterprise design and operation (production and coordination rules, exceptions list).

References

1. Russell L. Ackoff. *Ackoff's Best: His Classic Writings on Management.* Wiley, 1999.
2. Jan L. G. Dietz. *Enterprise Ontology: Theory and Methodology.* Springer, 2006.
3. Jan L. G. Dietz. *Architecture - Building strategy into design.* Academic Service, 2008.
4. John Dobson Guy Dewsbury. *Responsibility and dependable systems.* Springer, 2007.
5. Van Haren. *TOGAFTM The Open Group Architecture Framework A Management Guide.* Van Haren Publishing, 2005.
6. Jan A. P. Hoogervorst. *Enterprise Governance and Enterprise Engineering (The Enterprise Engineering Series).* Springer, 1 edition, February 2009.
7. Jan A. P. Hoogervorst and Jan L. G. Dietz. Enterprise architecture in enterprise engineering. *Enterprise Modelling and Information Systems Architectures*, 2008.
8. IEEE1471. Recommended practice for architectural description for software- intensive systems. Technical report, IEEE Computer Society, 2000.
9. Michael C. Jackson. *Systems Thinking.* Wiley, 2003.
10. Robert S. Kaplan and David P. Norton. *Strategy maps.* Harvard Business School Press, Boston, Mass., 2004.
11. Eberhardt Rechtin. *Systems Architecting of Organizations: Why Eagles Can't Swim.* CRC, 1 edition, July 1999.
12. Pedro Sousa, Artur Caetano, Andre Vasconcelos, Carla Pereira, and Jose Tribolet. *Enterprise architecture modeling with the Unified Modeling Language.* Idea Group Publishing, 2006.
13. Gerald M. Weinberg. *An Introduction to General Systems Thinking.* Dorset House Publishing Company, Incorporated, April 2001.