

BUSINESS RULES MANAGEMENT SOLUTIONS PROBLEM SPACE: SITUATIONAL FACTORS

Martijn Zoet, HU University of Applied Sciences Utrecht, Nijenoord 1, 3552 AS, Utrecht, Netherlands, martijn.zoet@hu.nl

Johan Versendaal, HU University of Applied Sciences Utrecht, Nijenoord 1, 3552 AS, Utrecht, Netherlands, johan.versendaal@hu.nl

Abstract

Business rules management solutions are widely applied, standalone or in combination with business process management solutions. Yet scientific research on business rules management solutions is limited. The purpose of this paper is to define the business rules management solution problem space. Using contingency theory and relational theory as our lens, we conducted a qualitative study on 39 business rules management solutions. The range of data sources included interviews and document analysis. From the qualitative study six situational factors have been defined to classify the business rules management solution space: 1) value proposition, 2) approach, 3) standardization, 4) change frequency, 5) n-order compliance, and 6) integrative power of the software environment. The six factors can be clustered in three structures 1) deep structure, 2) physical structure and, 3) organizational structure. The classification of the problem space provides a framework for the analysis of business rules management solutions.

Keywords: Business Rules, Business Rules Management, Problem Space, Situational Factors.

1 INTRODUCTION

Business rules management and business process management both study the management of activities and decisions. The difference between the two is the adopted viewpoint. Business process management (BPM) adopts an activity/resource viewpoint while business rules management (BRM) adopts a knowledge/guideline viewpoint (Zoet et al., 2011). The last decade an increased interest to integrate the two viewpoints has emerged in scientific as well as professional literature. Research to do so has been and is currently executed in the domain of business process and business rule formalization, classification, articulation, and technical interoperability (zur Muehlen & Indulska, 2010). We are in agreement with Rosca and Wild (2002) and Nelson et al. (2010) that a broader view of integrating business processes and business rules should be taken. Thus, not only focusing on the technical aspects but also connecting both problem spaces and management practices. BPM research already explicitly focuses on management practices and the definition of the business process management problem space (Bucher & Winter, 2010). However, research focusing on management practices for BRM is limited (Arnott & Pervan, 2005; Nelson et al. 2010; Rosca & Wild, 2002). Consequently, the problem space ‘business rules management solutions’ needs to be defined before connecting the two fields from a management perspective.

A business rule is (Morgan, 2002) “a statement that defines or constrains some aspect of the business intending to assert business structure or to control the behavior of the business.” A business rules management solution (hence BRMS) enables organizations to elicitate, design, manage and execute business rules and is a co-creation of eleven service systems (Zoet & Versendaal, 2012) namely 1) the monitoring service system, 2) the execution service system, 3) the deployment service system, 4) the verification service system, 5) the validation service system, 6) the design service system, 7) the improvement system, 8) the mining service system, 9) the cleansing service system, 10) the version service system, and 11) the audit service system. Each individual implementation of a BRMS is a specific instantiation of previous mentioned service systems.

BRMSs are commonly addressed as singular problem-oriented, meaning that a specific BRMS is designed to solve one specific problem (Liao, 2004; Wanger et al., 2002). Yet, previous research has shown that different BRMSs have a common design problem. A common design problem indicates that common problem classes, for which design solutions can be created, exists (Simon, 1970; Winter, 2011b). Winter (2011b) defines a problem class as a set of similar design problems. A problem space can contain one or more problem classes. For example, decision management and process guidance can be problem classes of the problem space BRMS. An instantiation of a specific problem class in a specific organization is defined as a design solution. In the BRMS problem space the design solution is a specific configuration of the earlier mentioned eleven service systems.

Both problem spaces and design solutions are subject to situational factors (Winter, 2011b). Situational factors describe the context in which an IS artifact or organization has to operate such that the deployed artifact fits the context of the environment. Research identifying situational factors is executed, among others, in software product management (Bekkers et al., 2008), business process management (Bucher & Winter, 2010) and, enterprise architecture (Klesse & Winter, 2007). Research focusing on situational factors affecting business rules in general and the BRMS problem space specifically, to the knowledge of the authors, is absent. This article extends the understanding of BRMSs by addressing the situational factors that characterize different problem classes. With these premises, the following research question is addressed: “*Which situational factors describe the design of a Business Rules Management problem space?*” Answering this question will help organizations better understand the design and management of BRMSs.

The paper is organized as follows. First, we start by looking at contingency theory and relational theory, which we consider the fundament for our research. After which the relationship between problem classes, design situations, and situational factors is explained. Section three, describes the collection and analysis of 39 BRMS implementations. After which the results of the data analysis, the identification of six situational factors to classify the BRMS problem s, are presented in section 4.

Section 5 theorizes and compares the results of our research to previous research. Furthermore the limitations and contribution to theory and practitioners are presented. We conclude and summarize our research in section 6.

2 LITERATURE REVIEW

The core proposition of contingency theory is that a fit between situational factors and organizational structure of an enterprise leads to performance while a mismatch leads to lack of performance; indicating that the effect of one variable by another depends upon a third variable (Donaldson, 2001). Empirical evidence supporting and rejecting this theory have both been found and therefore some scholars heavily criticize its validity (Pfeffer, 1997). Still the central idea that fit positively affects performance is accepted in the scientific community (Strong & Volkoff, 2010; Winter, 2011a). When constructing solutions, methods or information systems situational factors should be considered to achieve a proper fit between the constructed solution, method, information system and the environment in which they are applied. Situationality is the similarity or dissimilarity of two or more problem classes expressed in terms of situational factors. Relational theory (theory of networks) state that systems, organizations, artifacts are differentiated by reduction in degrees of freedom taking into account the different levels in which freedom can occur (Economides, 1996; Lin, 1999). Thus situational factors from a relation theory viewpoint reduce the degree of freedom of a problem class. Therefore problem classes can be viewed as the product of unique, relational ordered, situational factors.

To explain the difference between problem spaces, problem classes, design situations and situational factors we adopt the Chinese house example by Winter, see figure 1. The problem space depicted is building a *Chinese style house*. This problem space is divided into problem classes by situational factors. For example the foundation and framing of the house reduce the degree of freedom thereby specifying problem classes. Problem classes again can be further specified by means of situational factors, representing the different levels in which freedom can occur. If no further reduction in freedom can occur different problem classes for building the Chinese Style house have been defined. Each problem class now represents a design situation that can be built. The instantiation of the actual design situation itself is also influenced by situational factors. For example if the problem class *Chinese House A* states that the structure and roof of the house must be circular it doesn't state anything about the material used in the actual instantiation. This can differ per house build. House number one can be build with bricks while a second house can be build with wood. Material in this case is a situational factor influencing the actual construction of the house. Situational factors affecting the problem space are the minimal number of situational factors necessary to classify a specific problem class, which we define as the classification freedom of the problem space. Thus, situational factors reducing the freedom of a problem class exist in all instantiations of design situations whereas situational factors affecting solely design situations are not.

Design solutions addressing a specific BRMS problem space are a configuration of the earlier mentioned eleven service systems. A detailed explanation of each service system can be found in (Zoet & Versendaal, 2012). However to ground our research method a summary is provided here. To deliver the value proposition of a BRMS, business rule models need to be design. Before a model can be designed data sources need to be mined for information. Data sources can be sources such as human experts, documentation, laws, and regulation. The 1) mining service system contains processes, techniques and tools to extract information from various sources. In some cases the data sources have to be cleansed to accomplish the desired mining effect. Data that intervenes with proper mining or design activities is removed from a data source by the 2) cleansing service system. After cleansing and mining, the non-platform specific rule model is created within the 3) design service system. Additionally an 4) improvement system exists which contains processes, techniques, algorithm, and tools for optimization and impact analysis of the designed rule model.

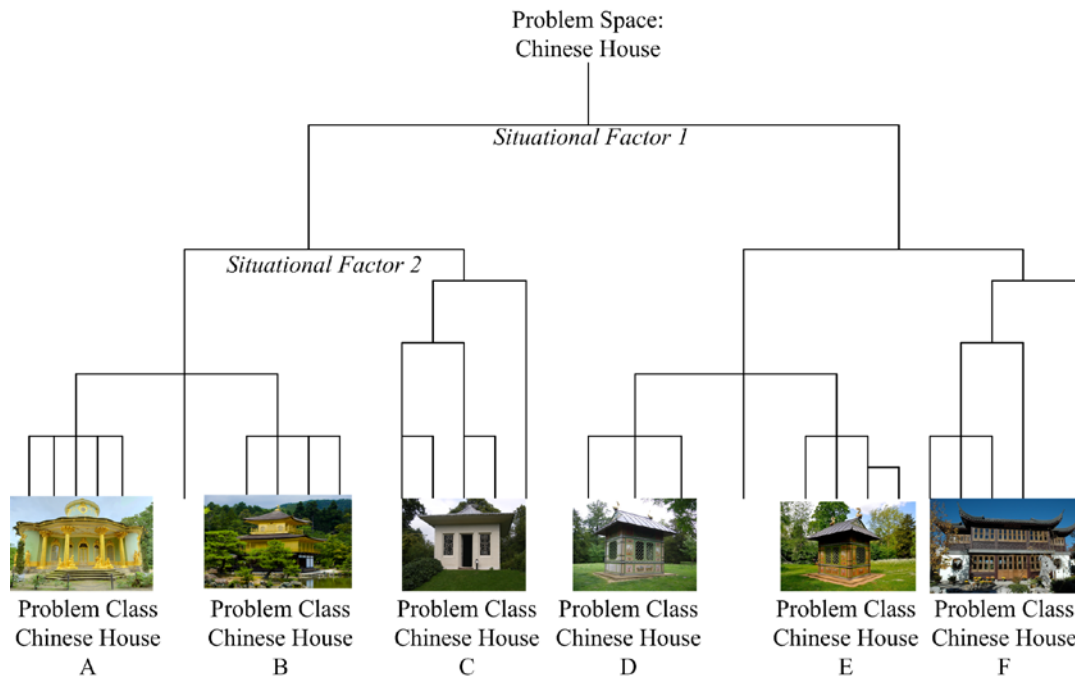


Figure 1. Chinese House Example

After the rule model is created it is checked for two types of errors: A) semantic / syntax errors and B) errors in its intended behavior. The first type of errors are removed from the rule model by the 5) verification service system; the latter by the 6) validation service system. The 7) deployment service system transforms the validated and verified models to a platform specific rule model. The platform which executes the business rules can be human or automated. A platform specific rule model can be source code, handbooks or procedures. Execution of business rules is guided by a separate service system: 8) the execution service system. It transforms a platform specific rule model into the value proposition it must deliver. Deployed business rules are monitored for proper execution. The 9) monitoring service system collects information from executed business rules and generates alerts when specific events occur. This information in turn can be used to improve existing rule models or design new rule models. All service systems provide output to two management service systems: 10) the audit service system and 11) the version service system. Data collected about realizing changes to specific input, output and other service system elements are registered by the audit service system. Examples of registered elements are: execution dates, rule model use, rule model editing, verification and validation. Changes made to the data source, platform specific rule models, non-platform specific rule models and all other input and output are registered by the version service system.

3 RESEARCH METHOD

3.1 Research design

The goal of this research is to identify the situational factors that characterize the BRMS problem space. To accomplish this goal a research approach is needed that can 1) identify situational factors, 2) identify similarities and dissimilarities between situational factors, and 3) identify the similarities and dissimilarities of situational factors between cases. The first two goals are realized by applying grounded theory. The purpose of grounded theory is to “explain with the fewest possible concepts, and with the greatest possible scope, as much variation as possible in the behavior and problem under study.” Grounded theory identifies difference and similarities by applying eighteen coding families. However, this does not provide a structured comparison of the identified situational factors across cases. A technique specifically engineered to inspect cases for similarities and differences is ordinal

comparison based on Mill's method of agreements and difference (Mahoney, 1999). Mill's method states that the cause of a phenomenon is the characteristic or combination of characteristics found in each case (Mill, 1906). Translated to our situation this means that the minimal set of situational factors needed to describe the BRMS problem space are the situational factors present at each BRMS. Therefore Mill's method in combination with grounded theory is adopted for this analysis.

3.2 Data collection

The concurrent data collection and analysis during the grounded theory study included the analysis of 63 project documents and approximately 18 hours of semi-structured interviews. In concurrence with the grounded theory methodology (Corbin and Straus, 1990) the interviewees as well as the projects have been selected based on concepts under investigation, their properties, dimensions and variations. The first selection within a grounded theory research is based on the phenomenon studied and a group of individuals, organizations or communities that best represent this phenomenon (Corbin and Straus, 1990). For example if one wants to study the work of nurses, one goes to a hospital or clinic. In our case we went to the Business Rules Platform the Netherlands; a community debating and discussing the need and use of business rules based services. From their 454 members we selected two organizations to start conducting interviews and collecting project documentation. The unit of analysis is a single BRMS, implying that one organization can contribute multiple units of analysis. To contribute cases, consultancy agencies, vendors or system integrators must have advised on or implemented multiple BRMSs, preferably in multiple industries. For all other organizations the criterion is that they implemented one or more BRMSs and preferably also applied changes to the specific solution over time. In total we analyzed 39 BRMSs, for details see table 1.

Industry	Number of BRMS
Financial	11
Medical	4
Transport	1
Government	19
Remainder	4
Total	39

Table 1. Number of BRMS analyzed per industry

Data for this study were collected through written documentation (vision documents, project documentation, internal communication, project presentations and evaluations), and semi-structured interviews with 15 informants at various organizations. Semi-structured interviews were conducted with four enterprise architects, six business rules architects, three business rules system architects and one subject matter expert from government and industry. The interviews on average took about 2,5 hours. During the interview sessions respondents were first asked to describe a specific BRMS based on the eleven service systems and their characteristics. During the second part they were asked to indicate changes over time for the same BRMS. The last part of the interview focused on changing specific implemented elements for a specific service system and asking respondents to indicate the impact on other service system elements. All interviews were recorded.

3.3 Data Analysis

Data analysis was conducted in several iterations following three cycles of coding namely (1) open coding, (2) axial coding, and (3) selective coding (Straus & Corbin, 1990) and one cycle of ordinal comparison, and narrative analysis. During the first cycle, text fragments, either individual words or sentences, have been classified as situational factors. Due to space limitations the complete matrix is not be added to the paper. A snapshot of the situational factors matrix has been added instead, see table 2. After open coding, axial coding has been applied. During axial coding relationships between categories must emerge. Relationships can be identified by applying eighteen coding families (Glaser 1978).

Text	Situational Factor	Inductive	Deductive
The roles needed for the execution of this project are: end users to validate the business rules, lawyers to validate business rules. Rule analysts to elicitate and the design the business rules. Testers to validate and veriflicated the business rules. Architects to validate the architecture principles.	End users	End User	
	Lawyers	Lawyer	
	Rule analyst	Rule analyst	
	Testers	Tester	
	Educators	Educator	
	Architects	Architect	
	Programmers	Programmer	
	Architects	Architect	
In this project the current business rule models, depicted in Microsoft Word and Oracle Policy Automation, are translated to The Decision Model.	Word	Software	
	The Decision Model	(1) Modeling Notation / (2) Non-Standard Modeling Language	
	Oracle Policy Automation	(1) Modeling Notation / (2) Non-Standard Modeling Language	
Our recurring propositions for BRMSs are self service processes, customized advice, scheduling and granting.	Value Proposition		Self Service Processes
	Value Proposition		Customized advice
	Value Proposition		Scheduling
	Value Proposition		Granting

Table 2. *Situational Factor Matrix*

This process requires inductive as well as deductive reasoning and data collection. Inductive reasoning has been applied to reason from concrete factors to general situational factors. For example, a project report from a government agency stated that two modeling notations are applied 1) The Decision Model Notation and 2) Oracle Policy Automation Modeling Language. During open coding both were coded as *situational factor*. Iterating between open coding and axial coding both were re-coded to *modeling languages*. Applying the eighteen coding families, the type family indentified a difference between *standard modeling language* and *non-standard modeling language*. Since both modeling languages are not (yet) an accepted standard both were re-coded to *non-standard modeling language*. Reasoning from general factors to case instantiations has been applied when respondents argued on specific situational factors occurring in multiple cases. For example, consultancy firm X stated that *value proposition* is a situational factor that affects a BRMS. For each case, the *value proposition* was described: 1) self service processes, 2) customized advice, 3) scheduling and 4) granting. Value proposition at first glance is a category that should emerge, iterating between open coding and axial coding. Therefore open codes were reviewed to indentify value propositions.

Next, all situational factors per individual case have been transformed to columns in an ordinal comparison table. An ordinal comparison table exists of mutual exclusive categories, in our case situational factors, that either are present (1) or absent (0), see table 3. Due to space limitations the complete ordinal comparison table could not be added to the paper, a snapshot has been added instead. The rows depict the cases analyzed. For each case the presence or absence of the situational factor has been depicted in the table.

Case	Situational Factor					
	Value Proposition	Development Philosophy	Rule Analyst	Architect	Change frequency	Nth Situational Factor
Financial Case I	1	1	1	0	1	0
Financial Case II	1	1	0	0	1	0
Financial Case III	1	1	1	0	1	1
Government Case I	1	1	1	0	1	0
Government Case II	1	1	1	1	1	0
Government Case III	1	1	0	0	1	0
Government Case IIII	1	1	1	1	1	1

Table 3. Ordinal Comparison Situational factors

4 RESULTS

In this section the six indentified situational factors of the problem space are presented, see figure 2. The situational factors have been structured along the dimensions of the ontological foundations of information systems framework originally proposed by Weber (1997) and extended by Strong and Volkoff (2010). The framework is divided into four sections: 1) deep structure, 2) organizational structure 3) physical structure and, 4) surface structure. Deep structure elements are subjects that describe real-world systems, their properties, states and transformations (Weber, 1997). Three situational factors affect the deep structure: 1) value proposition (VP), 2) approach (A) and 3) standardization (S). Organizational structures are the roles, control and organizational culture represented within organizations or within solutions (Strong & Volkoff, 2010). Two situational factors affect the organizational structure: 4) change frequency (CF) and 5) n-order compliance (NC). Physical structure elements describe the physical technology and software in which the deep structure is embedded (Weber, 1997). One situational factor affects the physical structure: 6) the integrative power of the software environment (IP). Surface structure elements describe the interface between the information system and the users. No situational factors have been indentified that affect the surface structure.

4.1 Reduction of Freedom: Deep Structure Situational Factors

The *first situational factor* is the 1) *value proposition* realized. This results in a reduction of freedom in terms of subjects modeled. This in turn results in a reduction of applicable processes and output subjects for each of the eleven service systems. A business rules analyst described the reduction of freedom as follows:

“When the application of the business rules must be able to guide business processes they must specify input constrains, output constraints and sometimes sequence. When the application of the modeled business rules must be able to make a decision they must specify condition and conclusion. The manner in which both are designed, veriflicated, validated and deployed differs. As well as the languages in which we model them; BPEL and OPA [modeling language].”

Analysis of the 39 BRMSs indicated a large number of different value propositions and corresponding subjects modeled. For example, guidance of process execution, guidance of documentation creation, granting, guidance of interactive web documents, monitoring of actions, decisioning, and configuration of personal advice to name a few. Collected data allows defining detailed subcategories of value propositions and subjects modeled. However, after debate we decided to define generic value propositions and not yet detailed subcategories.

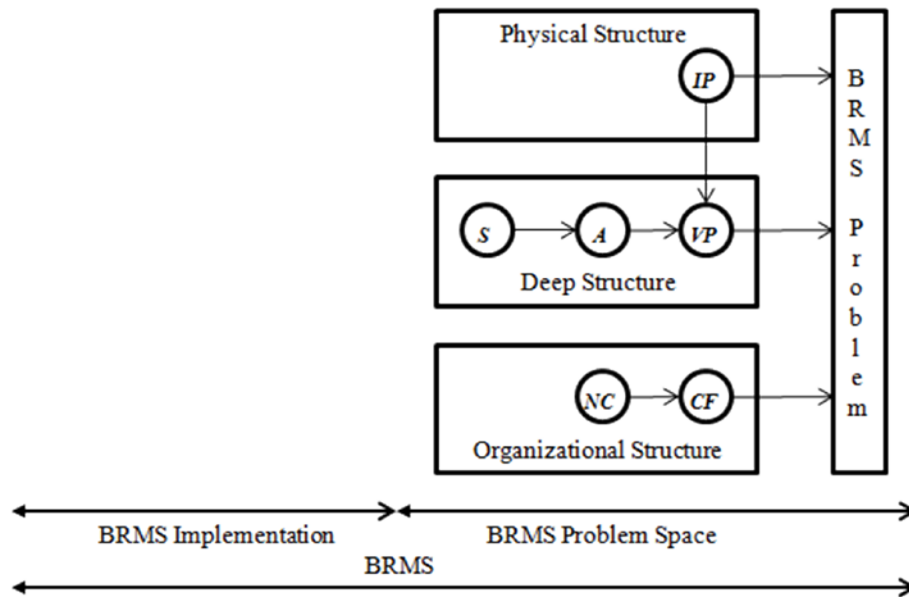


Figure 2. Casual Linkage Situational Factors

Therefore, in line with current literature, we define three different values for the value proposition situational factor: A) guidance (constraint), B) communication and C) decisioning. *Guidance elements* describe boundaries, borders or limits with regard to the behavior of business entities. This value proposition applies to a broad range of application areas and business rule statements. Business entities can be anything of value to the business for example databases, human resources, interaction elements and processes. The value proposition *communication* is realized by describing a business entity, its characteristics and/or relationships with other business entities. Definitions of actual business entities can be proposed: for example a driving license is an authorization for the bearer to drive a specified motorized vehicle. Therefore a driving license belongs to a person. *Decisioning* describes conditions evaluating business facts leading to a conclusion. The application of this statement depends on the application area. When applied to assess decisioning business rules are used to formulate a decision. However, when decisioning is applied to monitoring the business rules are used to formulate norms.

The *second situational factor* is 2) *approach*. The choice for a specific approach determines the model abstraction needed. This in turn results in a reduction of applicable service systems for the BRMS. Our analysis revealed three different values for approach: A) IT-oriented value, B) business-oriented value, and C) balanced value. The IT-oriented value emphasizes on enactable platform specific rule models. An enactable model is a model that can be executed by physical hardware or software. The output of the service systems are IT-related products such as technical design documents and functional design documents. On the other hand the business-oriented value focuses on realizing non-platform specific rule models. Business rule models realized with this value serve mainly for simulation and communication. The balanced value bridges both worlds. In the latter type the business units develop the non-platform specific rule model while the IT department translates the model to enactable platform-specific rule models. Nelson et al. (2010) identified the same values, however, viewing them through a maturity model lens. Where the IT value is classified as the lowest level and the business value / balanced value is classified as the highest. Although we identified BRMSs following the same shift in problem class, the other way around also is recognized. An architect and business analyst explain:

“Business Rules are the single point of knowledge within an organization. Only a limited number of business people maintain the business rules. The rules are directive for each action taken and every form of communication inside and outside the organization. In our case the information department might use the business rules as input but they do not create business rule model themselves. Long term strategy [5- 10 years] might allow this, but currently: no”

“We started our business rules approach at the product engineering department. When the process was mature enough at the business side we started to bridge the gap to the IT department”

The *third situational factor* is defined as 3) *standardization*. Analysis identified two different values: A) standardized modeling language or B) non standardized modeling language.

4.2 Reduction of Freedom: Organizational Structure Situational Factors

The *fourth situational factor* 4) *change frequency* of business rules affects the organizational structure of a BRMS. Change frequency indicates the number of times business rules change which we classify as A) low, B) medium and C) high. When the change frequency is high it is necessary to setup proper processes, roles, input and output for the audit service system and the version service system. When a business rule set never changes or almost never changes such a structure is not necessary, as described by an architect:

“We have multiple BRMSs in our organization. The business rules for insurance products change 70 times per two weeks. Here we have a very strict change process that exists of five formal steps [...] and a very strict version and audit policy. [...] We also apply business rules for specific events, checking these business rules haven’t changed the last 1,5 let’s say 2 years. This process does not have a strict and formal change process and versions aren’t saved”

The *fifth situational factor* is 5) *N-Order compliancy*. N-Order compliancy is a measurement to measure the number of actors between the enforcer and/or creator of the law/regulation/strategy and the actual implementation by means of business rule models. Only one role within organizations has the power (and knowledge) to provide 1st order compliancy: the role that defines the regulation. They can achieve this by translating the law into a business rules model themselves or by validating the model created by other roles. To achieve this in practice specific roles and control elements need to be added to the design, verification and validation service systems. In other situations 1st order compliance is not possible at all and the design, verification and validation service systems need to be designed in a different manner as this business architect explains:

“Most of our business rules are directly derived from regulation. Regulation created by lawyers at the ministries. This regulation is interpreted by our analysts and models are created. In the old situation these models supposed to be checked by our own lawyers however this check only existed on paper. In the current situation our analysts still transform the regulation to models. However our lawyers validate the models. It would be more convenient if our lawyers could do the translation. It would be perfect if the lawyers at the ministries and our lawyers together would do the translation.”

Another situational factor related to organizational structure is present in all cases, i.e. *Project Philosophy*. Project philosophy is the development philosophy the organizational unit follows. Values indentified during our analysis are A) agile, B) waterfall and a C) combination of both. After debating this situational factor we decided to remove it from our analysis. The rationale behind this decision is that every solution implemented in an organization has a specific project philosophy which is not unique for a BRMS.

4.3 Reduction of Freedom: Physical Structure Situational Factors

Physical structure is recognized as a separate structure. Still the situational factors indentified are highly coupled to the situational factors of the deep structure. This is consistent with the viewpoint expressed by Weber (1997) that the physical structure is the way in which the deep structure is mapped onto hardware and software. To support the different aspects of a BRMS multiple software functions are needed. These functions can be integrated into one software package or distributed across multiple software packages.

The *sixth situational factor* the 6) *integrative power of the software environment* is a measurement to determine the distribution of functions needed for the BRMS. Our analysis revealed two values: A) integrated and B) non-integrated. A software environment that is integrated provides software

functions for one or more service systems within one software package. A software package that delivers functions to support only one service system is a non-integrated software system. A business architect describes this functionality in practice:

“In general all functions needed for a BRMS can be loosely coupled. However, performance of specific tasks will be highly effect if specific functions are loosely coupled. Examples of such tasks are predictive analytics, simulation or high performance monitoring. In these cases software packages that integrate design, validation and improvement must be used to deliver the solution.”

5 CONTRIBUTIONS, LIMITATIONS AND DISCUSSION

From a research perspective our study provides a theoretical fundament for the BRMS problem space and configuration of underlying service systems. An important step since clusters have been defined that can be used to define situational methods, grammars and practitioners can better manage resources within business rules service systems. The contribution of our problem space framework for BRMS can be understood in relation to design research literature, ‘organization-enterprise system fit literature’ and ‘management fit literature’. Most authors start the design process with the identification of the relevant problem (Baskerville et al. 2009, Eekels & Roozenburg, 1991; Takeda et al. 1990). However, when taking into account situational factors, current research often does not focus on indentifying the problem space but rather the specific design implementation.

We consider Sia and Soh’s (2007) misalignment assessment framework, and Strong and Volkoff’s (2010) organization-enterprise system fit types. Sia and Soh’s (2007) propose a framework that predicts how organizations will resolve misfits in enterprise system configuration. The data analyzed by Sia and Soh (2007) is based on change requests for enterprise systems. Thus their framework measures misfit of the solution artifact deployed which is the information system. They apply three measurements (severity, frequency, and resolution) to externally imposed criteria and voluntarily acquired criteria (Sia & Soh, 2007). The criteria are also mapped on Wand and Weber’s ontological structure (Wand & Weber, 1995). In Sia and Soh’s (2007) framework one cannot distinguish between misfits caused by wrong assessment of the problem space and a wrong implementation of the solution artifact. Our framework allows us to do so.

Strong and Volkoff (2010) propose two fit types of ‘organization-enterprise system fit: 1) coverage fit and 2) enablement fit. Coverage fit is achieved by eliminating deficiencies and tailoring an information system through configuration and customization. Coverage fit affects the problem space as well as the solution artifact, since eliminating deficiencies happens in both. Enablement fit is a measurement solely for the solution artifact that is deployed since it measures the actual usage of the information system deployed. Strong and Volkoff (2010) fit types should be further analyzed to investigate the difference between coverage fit on the problem space level and the implementation level.

An example of a framework that proposes situational factors for a specific problem space is described by Henderson and Venkatraman (1993). The framework specifies four specific problem classes for the problem space business-it alignment. For each problem class they describe the limitation of freedom. For example, the role of top management is prioritizer when the problem class is service level alignment. This role changes to business visionary when the problem class changes from service level alignment to competitive potential alignment (Henderson & Venkatraman, 1993). They do not describe limitations for the solution artifact to implement the service level alignment in a specific organization. Our framework focuses on a different problem space but addresses fit in the same manner.

To accommodate different levels of situational factors we extent the enterprise system artifact proposed by Strong and Volkoff (2010) with an addition level, see figure 3. Our representation presents a view of a design artifact as the combination of four structures (physical, organizational, deep, and surface) on both the design problem level as well as the solution artifact level.

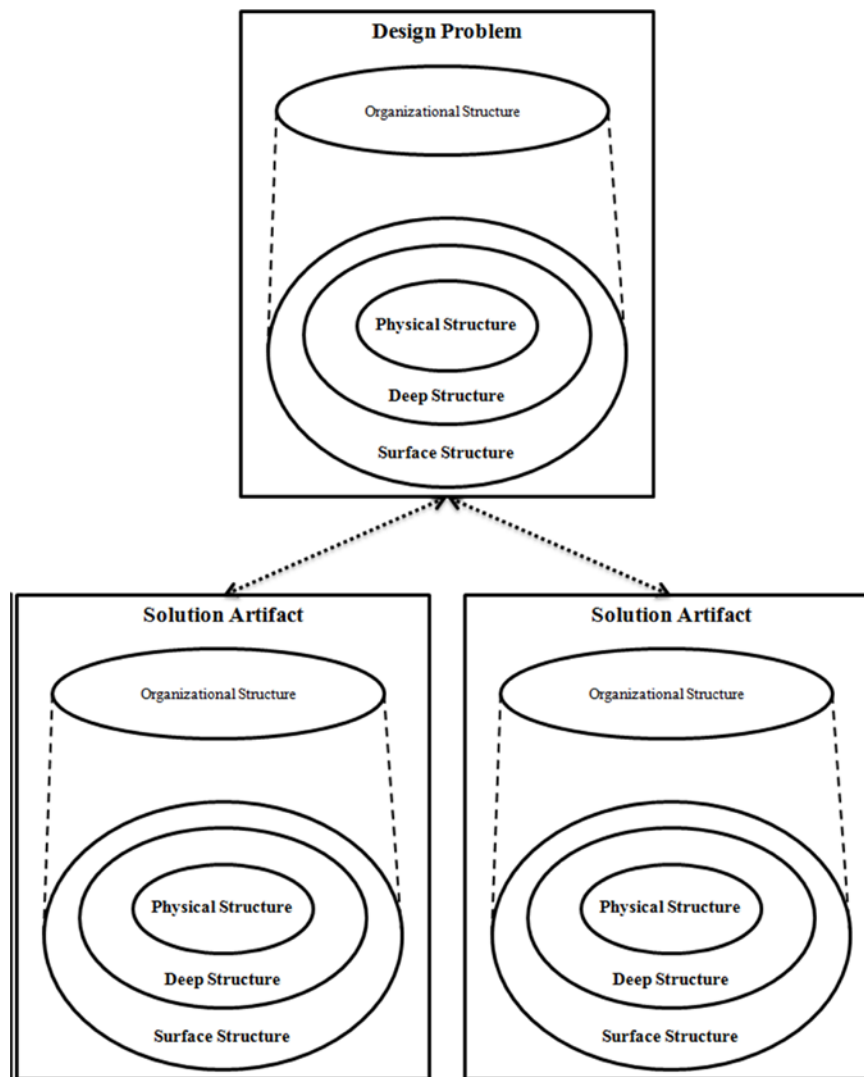


Figure 3. *Overview Adapted Ontological Foundations of Information Systems*

However each organization experiences further limitation of freedom through situational factors. These situational factors are depicted at the lower level, the solution artifact. Apart from the contribution to the business rules management knowledge base this also illustrates a different lens when applying contingency and relational theory. Both theories are mainly adopted to analyze and illustrate the representation of design solutions, thereby ignoring situationality of the problem space. We argue that more attention should be aimed towards properly indentifying, analyzing, and describing information system problem spaces.

From a practical perspective our study provides organizations and management within organizations with a diagnostic tool for identifying and describing their business rules management problem space. It offers a framework that can structure thinking about the solution to be implemented.

Several limitations may affect our results. The first limitation is the number and type of BRMSs analyzed. While we believe our study is representative of a large number of BRMSs, most solutions analyzed are implemented in organizations based in the Netherlands. This limits generalization. The second limitation is that a number of cases were provided by two consultancy firms. It could be argued that our study reflects a bias towards to the situationality the firms perceive when designing and implementing a BRMS. However, because our objective is to analyze the degree of freedom of the BRMS problem space and the consultancy firms have to deal with the situationality experienced by their customers, this does not significantly influence the results. Our study describes a BRMS problem space relying on induction and deduction. The only way to assess the generality of a theory is

through the use of deduction (Lee & Baskerville, 2003). Although deductive reasoning has been applied during this study it is only used within the analyzed cases. To further generalize the model a deductive validation outside the current units of analysis should be conducted; we note that such a deductive validation is outside the scope of this paper.

6 CONCLUSIONS

This research investigated the design factors of BRMSs with the purpose of developing a conceptualization of the BRMS problem space, and from this to identify specific BRMS problem spaces. To accomplish this goal, we conducted an analysis of situational factors using ordinal analysis to assess the minimal number of situational factors necessary to classify the BRMS problem space. This analysis revealed six situational factors, see figure 2: 1) value proposition, 2) approach, 3) standardization, 4) change frequency, 5) n-order compliance, and 6) integrative power of the software environment. Subsequently, analysis of the six situational factors using narrative comparison revealed three separate casual structures of situational factors 1) organizational structure, 2) deep structure and, 3) physical structure. Additionally our analysis also revealed a new conceptualization of the ontological foundations of information systems. In summary, our purpose in this paper was to study the minimal number of situational factors necessary to classify a BRMS problem class. Through coding, ordinal analysis and narrative analysis we have accomplished this purpose.

References

- Arnott, D., & Pervan, G. (2005). A critical analysis of decision support systems research. *Journal of Information Technology*, 20(2), 67-87.
- Baskerville, R., Pries-Heje, J., & Venable, J. (2009). Soft design science methodology. Paper presented at the Proceedings of the 4th International Conference on Design Science Research in Information Systems and Technology, ACM.
- Bekkers, W., van de Weerd, I., & Brinkkemper, S. (2008). The Influence of Situational Factors in Software Product Management: An Empirical Study. Paper presented at the 21th International Workshop on Product Management, Barcelona, Spain.
- Bucher, T., & Winter, R. (2010). Taxonomy of Business Process Management Approach. In v. Brocke & M. Rosemann (Eds.), *Handbook on Business Process Management*. Heidelberg: Springer.
- Donaldson, L. (2001). *The Contingency Theory of Organizations*: Sage, Thousand Oaks.
- Economides, N. (1996). The Economics of Networks. *International Journal of Industrial Organization*, 14(2), 675-699.
- Eekels, J., & Roozenburg, N. (1991). A methodological comparison of the structures of scientific research and engineering design: their similarities and differences. *Design Studies*, 12(1), 197-203.
- Henderson, J., & Venkatraman, N. (1993). Strategic alignment: a framework for strategic information technology management. *IBM Systems Journal*, 32(1), 472-484.
- Klesse, M., & Winter, R. (2007). Organizational Forms of Data Warehousing: An Explorative Analysis. Paper presented at the Hawaii International Conference on System Sciences, Hawaii.
- Lee, A., & Baskerville, R. (2003). Generalizing Generalizability in Information Systems Research. *Information Systems Research*, 14(3), 221-223.
- Liao, S.-H. (2004). Expert System Methodologies and Applications - A Decade Review from 1995 to 2004. *Expert Systems with Applications*, 28(1), 93-103.
- Lin, N. (1999). Building a network theory of social capital. *Connections*, 22(1), 28-51.
- Mahoney, J. (1999). Nominal, ordinal, and narrative appraisal in macrocausal analysis *American Journal of Sociology*, 104(4), 1154-1196.
- Mill, J. (1906). *A system of Logic*. London: Longmans Green.
- Morgan, T. (2002). *Business rules and information systems: aligning IT with business goals*. London: Addison-Wesley.

- Nelson, M. L., Peterson, J., Rariden, R. L., & Sen, R. (2010). Transitioning to a business rule management service model: Case studies from the property and casualty insurance industry. *Information & Management*, 47(1), 30-41. doi: 10.1016/j.im.2009.09.007
- Pfeffer, J. (1997). *New Directions for Organization Theory: Problems and Prospects*. New York: Oxford University Press.
- Rosca, D., & Wild, C. (2002). Towards a flexible deployment of business rules. *Expert Systems with Applications*, 23(4), 385-394. doi: 10.1016/S0957-4174(02)00074-X
- Sia, S., & Soh, C. (2007). An Assessment of Package-Organisation Misalignment: Institutional and Ontological Structures. *European Journal of Information Systems*, 16(5), 568-583.
- Simon, H. (1970). *The Sciences of the Artificial*. Cambridge, MA: MIT Press.
- Strong, D., & Volkoff, O. (2010). Understanding Organization-Enterprise System Fit: A Path to Theorizing the Information Technology Artifact. *MIS Quarterly*, 34(4), 731-756.
- Takeda, H., Veerkamp, P., Tomiyama, T., & Yoshikawa, H. (1990). Modeling design processes. *AI Magazine*, 11(1), 37-48.
- Wand, Y., & Weber, R. (1995). On the deep structure of information system. *Information Systems Journal*, 5(1995), 203-223.
- Wanger, W., Otto, J., & Chung, Q. (2002). Knowledge acquisition for expert systems in accounting and financial problem domains. *Knowledge-Based Systems*, 15(8), 439-447.
- Weber, R. (1997). *Ontological Foundations of Information Systems*. Australia: Coopers and Lybrand and the Accounting Association of Australia and New Zealand.
- Winter, R. (2011a). Design Solution Analysis for the Construction of Situational Design Methods. Paper presented at the ME 2011, Paris.
- Winter, R. (2011b). Problem Analysis for Situational Artefact Construction in Information Systems. In A. Carugati & C. Rossignoli (Eds.), *Emerging Themes in Information Systems and Organization Studies*. Berlin: Springer-Verlag.
- Zoet, M., & Versendaal, J. (2012). Business Rules Based Perspective on Services: A Mixed Method Analysis. Manuscript submitted for publication.
- Zoet, M., Versendaal, J., Ravesteyn, P., & Welke, R. (2011). Alignment of Business Process Management and Business Rules. Paper presented at the European Conference on Information Systems (ECIS).
- zur Muehlen, M., & Indulska, M. (2010). Modeling languages for business processes and business rules: A representational analysis. *Information Systems*, 35(4), 379-390. doi: 10.1016/j.is.2009.02.006