

# Design of Situational Artefacts—Conceptual Foundations and Their Application to IT/Business Alignment

Robert Winter

**Abstract** IT/business alignment is consistently rated as the #1 problem for CIOs of all kinds and sizes of companies. Although IS research has proposed a plethora of hypotheses and artefacts to explain and approach IT/business alignment problems, the large variety of goals and context factors seem to prevent effective solutions: Since companies face diverse challenges in achieving a high degree of IT/business alignment, a universal “one size fits all” approach seems not to be appropriate. IT/business alignment is therefore regarded as a class of relevant, important design problems that requires a situational solution approach. The conceptual foundations of situational problem analysis and artefact design are summarized, and a generic approach is proposed that transforms problem analysis into artefact design. The approach is then applied by decomposing the IT/business alignment problem into tangible qualities for business, IT systems, and IT governance. An explorative survey among 174 professionals is used to identify four clusters of IT/business alignment problems which each represent the current state according to certain qualities and also the priorities for future development. Using the proposed situational solution approach, four reference solutions are outlined.

## 1 Introduction

IT/business alignment has constantly been among the top priorities for IT executives (Luftman 2005; Luftman et al. 2006, 2009; Luftman and Kempaiah 2008). On the one hand, the IT/business alignment phenomenon needs to be better understood. On the other hand, artefacts need to be constructed on the foundation of that understanding which enhance or extend IT/business alignment. We concentrate on the latter research perspective. Our analysis of related work, i.e. existing methods and models

---

R. Winter (✉)

Institute of Information Management, University of St. Gallen, Müller-Friedberg-Strasse 8,  
9000 St. Gallen, Switzerland  
e-mail: Robert.Winter@unisg.ch

for IT/business alignment, in Sect. 2 shows that different problem contexts (e.g. industry, company size) and different design objectives (e.g. improving transparency, reducing inconsistencies, improving flexibility) are usually not considered. As diverse problem situations require different solution designs, design research should provide situated or at least adaptable solutions instead of “one-size-fits-all” artefacts and might thereby enhance or extend their utility. As a precondition for constructing such solutions, however, precise specifications of relevant IT/business alignment situations are needed. From that knowledge about problem situations, important aspects of proposed solutions can then be derived.

The contribution of this chapter is to propose a transformation between situational problem analysis and situational artefact construction. Since IT/business alignment is a diverse and widely unsolved problem, this class of design problems is used to instantiate the proposed approach, i.e. to identify problem situations and derive situational solution artefacts for IT/business alignment.

The remainder of this chapter is structured as follows: Sect. 2 presents and discusses related work on IT/business alignment. The often “fuzzy” understanding of IT/business alignment is operationalized by considering qualities of IT systems, business, and IT governance. In Sect. 3, we present conceptual foundations of situational problem analysis and situational artefact design. Based on a broad empirical analysis, four IT/business alignment situations are presented in Sect. 4. The transformation of situational problem analysis into appropriate solution artefacts for IT/business alignment is outlined in Sect. 5. The findings are discussed, and an outlook on future work is given in Sect. 6.

## 2 Related Work on IT/Business Alignment

The framework proposed by Henderson and Venkatraman (1999) is widely applied and is generally considered to be a key reference alignment model. It contains four components needed for alignment assessment, viz. (1) business strategy, (2) IT strategy, (3) organizational infrastructure and processes, and (4) IT infrastructure and processes. The interactions of these four components along two dimensions, strategic integration and functional integration, form the state space of the model. Neither do the authors propose problem solution artefacts, nor do they consider situational differences of IT/business alignment problems.

Luftman (2000) develops a strategic alignment measurement tool based on detailed maturity models covering six distinct areas: (1) communications, (2) competence/value, (3) governance, (4) partnership, (5) technology scope, and (6) skills maturity. This work was later extended in Sledgianowski and Luftman (2005) into the Strategic Alignment Maturity Assessment (SAMA). While an assessment tool can be considered as an artefact that supports problem solution, Luftman’s approach has no situational characteristics.

Aier and Winter (2009) propose what they call *virtual decoupling* to achieve IT/business alignment, i.e. an architecture-centric approach that separates the external view on architecture from its particular implementation. In doing so, the level of complexity is significantly reduced, thus enabling a more rational and less myopic approach to managing the IT/business alignment. Aier and Winter do however not consider situational differences of IT/business alignment problems.

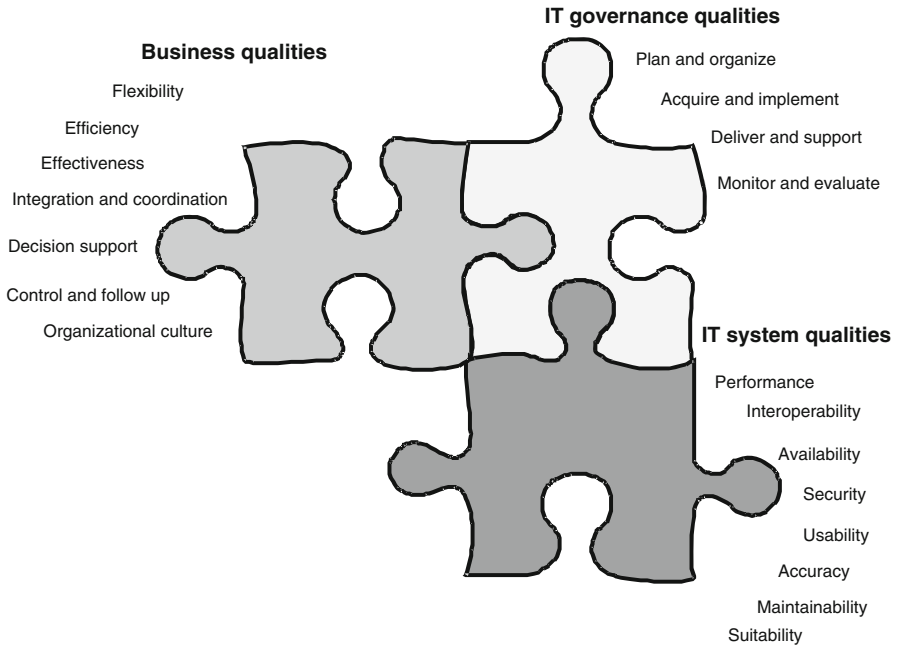
Plazaola et al. (2007) attempt to address strategic IT/business alignment by enterprise architecture management. They propose a meta model and explain the criteria as well as the process for connecting Luftman's strategic alignment measures (Luftman 2000) with Zachman's Enterprise Architectural Framework (Zachman 1987). However, the proposed meta model does not include any guidelines for how to apply it in particular enterprises and adjust it to the specific problem setup.

Based on an empirical study, Baumöl (2006) argues that IT/business alignment can only be achieved by using a systematic change method. She proposes such a method, called Change Method Engineering, which is based on empirical findings of critical IT/business alignment success factors. The approach is based on a taxonomy of transformation problems and thus can be considered as situational. The situations are however defined regarding general transformation and not specifically regarding IT/business alignment.

Although Baumöl (2006) at least attempts to provide a situational solution to IT/business alignment, all related work fails in addressing specific situational characteristics of IT/business alignment by appropriate solution artefacts. Moreover, related work exhibits a wide array of more or less fuzzy definitions of the IT/business alignment phenomenon. In order to specify IT/business alignment in a way that allows for systematic analysis and solution construction, we refer to an approach developed at KTH Stockholm. While Johnson and Ekstedt (2007) propose a set of measurable IT system qualities, Gammelgård et al. (2006) propose a set of measurable business and IT governance qualities that is aggregated later e.g. by Gustafsson et al. (2008). Figure 1 illustrates the conceptual analysis framework for IT/business alignment. In particular, the jigsaw puzzle metaphor is intended to convey that no single set of qualities has primacy over any other. IT/business alignment arises in the interaction of all the qualities and can be achieved only by adjusting all of them so that they fit each other. A comprehensive documentation of the qualities can be found e.g. in Saat et al. (2011).

### 3 Situational Problem Analysis and Situational Artefact Design

Design Science Research is a research paradigm that has been, among other application domains, successfully deployed to Information Systems (IS). In the following, we will use DSR to abbreviate Design Science Research in Information Systems. At its core, DSR is about the rigorous construction of useful IS artefacts, i.e. constructs, models, methods, or instantiations (March and Smith 1995). Hevner et al.



**Fig. 1** A conceptual view of an IT/business alignment operationalization. (Saat et al. 2011)

(2004) generalize constructs, models, methods and instantiations as “technology-based solutions to important and relevant business problems”. Design problems in organisations are generically defined as “the differences between a goal state and the current state of a system” (Hevner et al. 2004).

Besides of being important and relevant, the design problem—and hence the proposed design solution (= DSR output artefact)—should be sufficiently general. For Hevner et al. (2004), generality is one of three quality criteria of an DSR artefact. Baskerville et al. (2009) demand a design research artefact to “represent [...] a general solution to a class of problems”. In the following, we will therefore assume that DSR results are generic (and not specific) IS artefacts which are useful for solving a class of design problems.

The two research goals of generality and utility should be differentiated. In their research on reference modelling, Becker et al. discuss the *reference modelling dilemma*: “On the one hand, customers will choose a reference model that [...] provides the best fit to their individual requirements and therefore implies the least need for changes. On the other hand, a restriction of the generality of the model results in higher turn-over risks because of smaller sales markets” (Becker et al. 2007). This dilemma is not only apparent in reference modelling, but also exists for other general solutions to classes of design problems (e.g. methods). In a very simple form, it can be formalized as

$$U^*g = c$$

where  $U$  denotes an artefact's utility from a single organisation's perspective,  $g$  denotes its generality, and  $c$  is a constant (Winter 2011). With increasing generality, the individual utility of a solution for solving a specific design problem decreases—and vice versa. The sum of individual utilities is however increasing when solutions are specified on a higher level of genericity—but individual organizations might not be interested in this “general utility”.

As a solution to the reference modelling dilemma, Becker et al. (2002, 2007) propose adaptation mechanisms that instantiate a generic reference model according to the specific design problem at hand. We hold a larger view, i.e. refer to all four artefact types identified by March and Smith (1995) and hence designate the extension of generic artefacts by adaptation mechanisms as *situational artefact construction* (SAC). In addition to situational reference modelling (e.g. Becker et al. 2007), situational artefact construction has also been investigated for methods (situational method engineering, see e.g. Bucher et al. 2007).

As SAC allows the researcher to develop artefacts which are adaptable to different design problems within a problem class, a crucial decision during the construction phase is to delineate the range of addressed design problems (i.e. to specify the design problem class) and to understand the relevant design situations within this class. If a design problem class is understood as a set of “similar” design problems, a design situation can be understood as a subset of design problems which are even more similar. Depending on the degree of generality, a design problem class can be partitioned into few, very generic design situations or a larger number of (different) design situations of lesser generality.

Winter (2011) extended the initial procedure proposal of Bucher and Klesse (2006) for situational problem analysis by differentiating more components and assuming that, in general, only adaptable artefacts on a certain level of generality are constructed:

1. A rough idea about the delineation of the design problem class is developed. Results of this step are definitions, a description of the system under analysis and an idea about design goals for this class of design problems.
2. A literature analysis is conducted in order to identify potential contingency factors for that class of design problems.
3. A field study is conducted in order to analyze design problems of that class in practice. As a result, the list of potential contingency factor candidates is reduced to a smaller set of relevant “design factors”. Design factors might be aggregates of several contingency factors that need to be semantically interpreted.
4. The design problem class is redefined by specifying value ranges for the design factors. This means that “outliers” are ignored from further analysis in order to create homogeneous problem sets and subsets.
5. Those field study data of observations which still belong to the redefined design problem class, are used to calculate ultrametric distances between specific design problems. The calculation is based on certain “similarity” metrics—usually the Euclidian distance with regard to the observed values of design factors.

6. A useful level of solution generality is determined. Usually clustering errors related to the number of clusters are used for this analysis.
7. Using the desired solution generality, the resulting design situations are specified. The situations should not only be specified formally (by value ranges of the design factors), but also should be interpreted semantically (“design problem types”).

In Winter (2011), the discovery of four design problem types for data warehousing service provision is used as an illustrative example for steps (1) through (7).

Following this approach, the design problem class can be sufficiently analyzed to allow for the systematic development of solution artefacts. Figure 2 depicts an idealized graphical visualization of ultrametric distances between design problems (C1...C33) within a design problem class C. An ultrametric distance graph supports an intuitive understanding of how closely related (or how different) design problems in a class are Winter (2011). For example, the ultrametric distance between C11 and C15 is much smaller than that between C11 and C6 (i.e., C11 is more similar to C15 than to C6) because the common solution to C11...C15 is of lesser generality than a common solution to C1 . . . C15.

It also becomes intuitively clear that solution artefacts can be constructed on different levels of generality—the fewer artefacts are to be constructed, the higher their generality has to be. For the exemplary design problem class whose ultrametric distances are illustrated by Fig. 2, nearly any number of solution artefacts between 1 (one “one size fits all” generic solution for C) and 33 (one specific solution for every single design problem Cxx) could be constructed.

For the exemplary design problem class illustrated in Fig. 2, we assume that the analysis yields a desired generality level of  $G_c$ . For  $G_c$ , the ultrametric distance graph

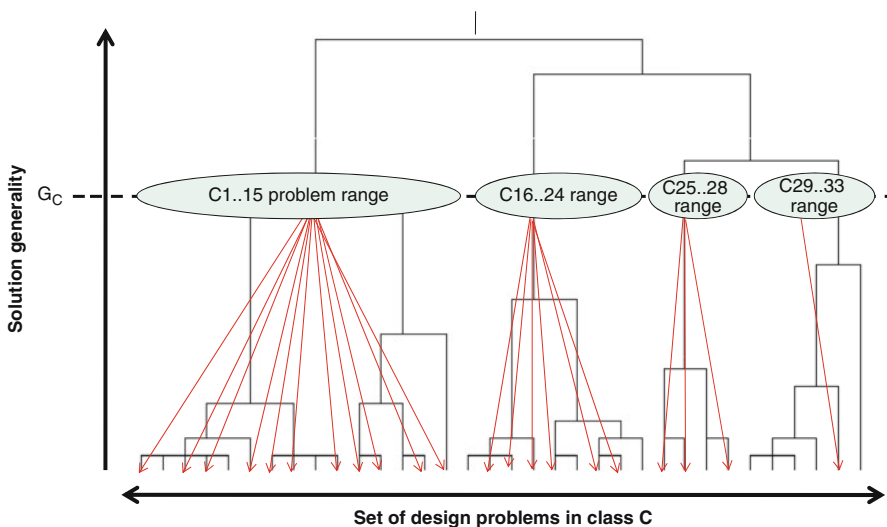


Fig. 2 Problem class decomposition into design situations. (Winter 2011)

yields four design situations (one for problems  $C1...15$ , one for problems  $C16...24$ , one for problems  $C25...28$  and one for problems  $C29...33$ ). Instead of developing only one “one size fits all” solution for problem class  $C$ , four solution artefacts would need to be constructed that cover the respective design situation. In order to provide a solution to one of the concrete design problems within  $C$ , the respective situational solution artefact would have to be adapted (directed arcs in Fig. 2). The graphical representation shows that such an adaptation would be much more targeted and much less costly than having to adapt a completely generic solution artefact for problems of type  $C$ .

The technique presented in Winter (2011) groups design problems into design situations that are homogeneous with regard to certain *design factors*, while separating design problems that are heterogeneous regarding such design factors. The design factors are not part of the initial design problem or problem class definition, but are instead the result of a principal component analysis on the problem description data. For typical design problem classes, a number of four to eight design factors are identified which explain the variance of the problem descriptions sufficiently. Every problem description can be interpreted as a point in an  $n$ -dimensional space which is implied by the  $n$  discovered design factors. A cluster analysis is then applied to identify  $m$  design situations which group these points into clusters of high intra-cluster homogeneity and inter-cluster heterogeneity.

An important DSR step is to semantically interpret the identified  $n$  design factors (= principal components of the problem data set) and  $m$  design situations (= clusters in the  $n$ -dimensional design factor space). For that purpose, it is often helpful to find out which subset of design factors characterizes a cluster best, i.e. which factor values are particularly high or low in a cluster and have a small deviation.

Understanding the problem-oriented meaning of design factors and design situations is essential for the construction of respective solution artefacts: Elementary problem solution fragments can be interpreted as “movements” in the 1-dimensional or 2-dimensional space implied by 1 or 2 relevant design factors, and complete problem solutions can be interpreted as  $p$ -dimensional vectors in the space implied by  $p$  relevant design factors for the respective solution ( $p \leq n$ ).

Summing up, the following procedure is proposed for situational solution design (and complements the situational problem analysis procedure):

8. By linking back analysis results to the characteristics of the underlying design problem descriptions, design factors (= result of principal component analysis) and design situations (= result of agglomerative cluster analysis) of the design problem class are interpreted semantically. This step is basically a more detailed and solution driven analysis than step 7.
9. Those combinations of design (problem) situations and 1 or 2 design factors are identified where the selected design factors represent best an elementary problem solution fragment. For example, if “widespread usage of concept X within the organization” is identified as a design factor and two design situations differ significantly with regard to this design factor (cluster A = low values, cluster

- B = high values), then “extend use of concept X” would be a problem solution fragment that effects a “movement” from cluster A to cluster B.
10. By linking desired properties to design factors, ideal problem solutions are related to the n-dimensional system. If many observations in the data set have already reached such desired state(s), there might be even one (rarely more) “design situation(s)” where the problem has already been solved—i.e. there might have been cluster(s) identified that represent(s) ideal problem solutions. Normally, there will be no such “problem already solved” clusters.
  11. For every design situation (except “problem already solved” clusters) and every ideal problem solution, all applicable solution fragments from (9) are aggregated into a solution path. Semantically, the activities which are represented by elementary “movements” in (9) are aggregated into an activity set. This activity set is usually not only related to 1 or 2 design factors, but involves many or even all identified design factors. If there are too many design situations and ideal problem solutions to enumerate this step, the most relevant “paths” from as-is to to-be can be extracted from survey data as long as to-be specifications—and not only as-is specifications—are including in the analysis.

#### **4 Problem Analysis: Four Types of IT/Business Alignment Problems**

This chapter summarizes Saat et al. (2011) who present findings of an online survey whose aim was, among others, to identify design situations for IT/business alignment. The survey was distributed among professionals in Sweden, Switzerland, Austria and Germany. Out of 1,105 invitations sent, 92 emails bounced, 339 persons started and 174 persons completed the survey. The online survey was active for ten days (Sept. 11–21, 2009). The survey included a final question regarding the respondent’s confidence with his or her answers. Twelve persons stated weak confidence so their answers were not further considered. A total of 162 completely filled in surveys are subjected to the following analysis. Figure 3 describes the dataset in detail.

The survey is comprised of four parts: Part one contains questions regarding the background of the respondents such as industry, country, and company size as well as the respondent’s role and experience. The second part of the survey has two sections. The Sect. 1 contains questions addressing enterprise architecture use for IT/business alignment and the importance and perceived maturity of IT/business alignment at the respondent’s company. Section 2 contains more detailed questions related to IT/business alignment and the positioning of the IT department within the respondent’s company. The third part of the survey addresses the qualities regarding IT, business, and IT governance as presented in Sect. 2. For each quality one statement was posted and the respondents were asked to mark the actual (as-is) situation (degree of realization) and desired (to-be) situation (importance for future realization) on a five-point Likert scale (where 1 equals very low, 2 equals low, 3 equals medium, 4 equals high, and 5 equals very high). The final part of the survey contains a



Industry	Count	Percent
Financial Industry	49	30.25
Utility	23	14.20
Manufacturing	17	10.49
Other	13	8.02
Transportation	12	7.41
Telecommunication	10	6.17
Software industry	8	4.94
Public sector	7	4.32
Defense/Military	7	4.32
IT/Management-Consulting	4	2.47
Healthcare/Pharmaceutical	3	1.85
Engineering/Architecture	3	1.85
Academia/Research	2	1.23
Tourism	2	1.23
Retail	2	1.23

Size of Comany	Count	Percent
<100	15	9.26
100-1.000	32	19.75
1.001-5.000	33	20.37
5.001-10.000	15	9.26
10.001-25.000	28	17.28
>25.000	39	24.07

Country	Count	Percent
Austria	4	2.47
Finland	1	0.62
Germany	28	17.28
Sweden	73	45.06
Switzerland	56	34.57

Area of Activity	Count	Percent
Business Management	28	17.28
Business Unit	22	13.58
IT Department	52	32.10
IT Management	60	37.04

n = 162

Fig. 3 Data set description. (Saat et al. 2011)

question regarding the respondents’ confidence regarding the answers as well as the possibility to submit questions and feedback to the authors.

According to the proposed procedure model, the data is first examined by exploratory factor analysis (Thompson 2004) in order to identify design factors with a large explanatory power. Second, these factors are used as input for a hierarchical agglomerative clustering procedure (Härdle and Simar 2003) that groups the problem descriptions represented by the respondent’s data into a small number of design situations with different characteristics.

The sampling adequacy is evaluated using the Kaiser-Meyer-Olkin (KMO) measure. In the case presented, the KMO is 0.918, which Kaiser and Rice (1974) regard to be “marvelous”. The Kaiser-Guttman criterion (Kaiser and Dickman 1959) and the elbow criterion (Cattell 1966) is used to obtain a desired number of factors. According to Kaiser and Dickman (1959) the number of factors to be extracted should be equal to the number of factors with eigenvalues bigger than one. The elbow criterion can be satisfied using a scree test as proposed by Cattell (1966). For the data set at hand, the optimal number of design factors is computed to be three.

Principal component analysis is used to extract the factors. The goal of this approach is to identify few but independent factors that reduce the items according to common characteristics. Table 1 depicts the result of the factor analysis. In order to better understand and interpret the factors, the component matrix is rotated using the Varimax method and Kaiser normalization (Kaiser 1958).

An item is assigned to a factor according to its factor load. The factor load is calculated using the correlation of a variable to the factor that was extracted from the data. In order for an item to be assigned to a design factor, the respective factor load has to be the highest value for this item and also has to exceed 0.5 (Härdle and Simar 2003).

**Table 1** Rotated component matrix. (Saat et al. 2011)

Cat.	Item	Factor		
		1	2	3
BUS	Decision support	0.753987	0.211965	0.188453
ITG	Acquire and implement	0.714991	0.186588	0.225826
ITG	Plan and organize	0.677433	0.224612	0.277512
BUS	Control and follow up	0.674663	0.096948	0.268977
BUS	Integration and coordination	0.637061	0.279205	-0.019899
BUS	Flexibility	0.636681	0.24160	-0.235255
BUS	Effectiveness	0.549010	0.483476	0.04557
ITS	Performance	0.051344	0.776337	0.196293
ITS	Availability	0.127147	0.69881	0.167507
ITS	Suitability	0.318333	0.654633	0.142230
ITS	Usability	0.275959	0.651576	-0.073737
ITS	Accuracy	0.184559	0.632516	0.338447
BUS	Efficiency	0.512070	0.537886	-0.002942
ITS	Maintainability	0.417643	0.516266	0.120234
ITS	Security	-0.056182	0.159936	0.768478
ITG	Deliver and support	0.373640	0.321855	0.631850
ITG	Monitor and evaluate	0.531410	0.061938	0.620340

*BUS* Business quality, *ITS* IT systems quality, *ITG* IT governance quality

Design factor 1 *Business-driven planning* consists of business and IT governance concerns. Items such as decision support, control and follow-up, business flexibility, effectiveness, integration and coordination, acquire and implement as well as plan and organize load on this factor. Participants (and hence IT/business alignment problems) scoring high in this factor emphasize strong business orientation and a strong IT organization.

Design factor 2 *System quality orientation* subsumes high priority of performance, availability, usability, accuracy, maintainability and suitability of the used IT systems. Therefore, companies (and hence IT/business alignment problems) scoring high in this factor focus on delivery of high-class and effective delivery of IT services to users.

Design factor 3 *Compliance focus* consists of the three items systems security, monitor and evaluate, and deliver and support. Common characteristics can be found in striving for a high degree of security and continuity of systems and business processes.

In order to identify as-is design situations within the data set, hierarchical cluster analysis is applied to the factor values calculated above. The fusion algorithm used is the Ward algorithm using Euclidian squared distance. This course of analysis creates distinct clusters by minimizing the internal cluster variance. This is the most popular approach for hierarchical clustering (Hair et al. 2006). The analysis results in four distinct design situations as presented in Fig. 4. The table presents the factor load for each cluster and the web diagram shows a graphical representation of these values.

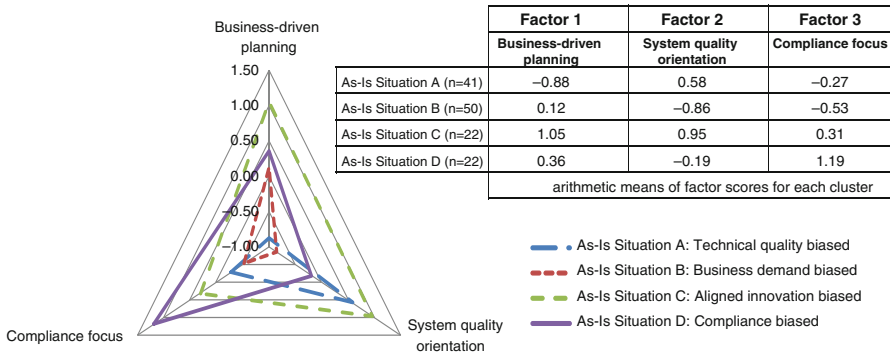


Fig. 4 IT/business alignment design situations. (Saat et al. 2011)

## 5 Artefact Design: Four IT/Business Alignment Solutions

In the preceding chapter, a semantic interpretation of the three design factors (step 8) has already been provided:

1. *Design factor 1 “Business-driven planning”* means that there is a strong (or weak) business orientation and a strong (or weak) IT organization.
2. *Design factor 2 “System quality orientation”* means that there is a strong (or weak) focus on delivery of high-class and effective IT service delivery to users.
3. *Design factor 3 “Compliance focus”* means that there is a high (or low) degree of security and continuity of systems and business processes.

The four identified IT/business alignment design situations within the analyzed sample can be then interpreted as follows:

- *Design situation A: technical quality biased*  
The first situation scores highest on the factor *system quality orientation*. Business planning issues are not regarded as very relevant. The degree of IT/business alignment is regarded to be rather low.
- *Design situation B: business demand biased*  
Business-driven planning is the most important issue for situation B. Technical quality is regarded much less relevant.
- *Design situation C: aligned innovation biased*  
Situation C delivers equally high scores for business and the IT-driven dimensions, while also considerably incorporating compliance aspects. It can be considered as a rather advanced group where a significant amount of IT/business alignment problems have already been addressed.
- *Design situation D: compliance biased*  
This situation is quite compliance focused and also somewhat business oriented. It is characterized by high perceived IT/business alignment. IT quality issues are regarded not very relevant.

Step 9 of the proposed procedure brings design situations together with design factors that seem to be of particular interest. The following combinations are obvious:

- *Design situation A:* Design problems represented by cluster A have the worst scores for design factor 1 and medium scores for design factor 3. They score well for design factor 2. The most interesting improvement linkages are to design factors 1 and 3.
- *Design situation B:* Design problems represented by cluster B have the worst scores for design factor 2, bad scores for design factor 3 and medium high scores only for design factor 1. There seems to be a lot of room for improvement in this cluster 2. The most interesting improvement linkage seems to be design factor 2.
- *Design situation C:* Design problems represented by cluster C have high scores for both design factors 1 and 2; Their scores for design factor 3 are medium. There seems to be not too much to improve in this cluster 2. If at all, the most interesting improvement linkage is to design factor 3.
- *Design situation D:* Design problems represented by cluster D score by far highest for design factor 3. Their scores for design factors 1 and 2 are medium. If at all, the most interesting improvement linkages are to design factors 1 and 2.

The desired “movements” are:

- *Design situation A:* IT/business alignment can be enhanced by (a) a stronger focus on business, (b) a more professional IT organization and (c) more emphasis on security and continuity. These measures promise strong effects on IT/business alignment.
- *Design situation B:* IT/business alignment can be enhanced by better and more effective IT service delivery. This measure promises strong effects on IT/business alignment.
- *Design situation C:* Although both the business and the technical perspective are quite mature in situation C, IT/business alignment could be enhanced by an increased focus on security and continuity of systems and business processes. Effects will be small.
- *Design situation D:* While overarching issues like security and continuity have quite a high maturity in situation D, both (a) business focus, (b) the IT organization’s professionalism and (c) IT service management could further be enhanced. Effects will be medium.

Step 10: Since all IT/business alignment qualities are regarded as equally important and desirable and because no explicit tradeoffs between qualities are evident, the “ideal solution” would be close to design situation C—maybe with a little more emphasis on overarching issues like security and continuity.

Since a comparatively small number of only three design factors has been identified in the sample, there is no need to aggregate different problem solution fragments into a complex problem solution (step 11). Based on the assessed effect of measures, the following solution artefacts would result from our analysis:

1. If IT/business alignment deficits result from a technical bias, mainly business qualities in the fields of decision support, acquire and implement, plan and organize, control and follow up, integration and coordination, flexibility and effectiveness need to be addressed. These activities should be accompanied by addressing IT governance qualities like security, delivery and support, as well as monitor and evaluate.
2. If IT/business alignment deficits result from a business demand bias, mainly IT qualities like performance, availability, suitability, usability, accuracy, efficiency and maintainability should be addressed.

The scales used for the characterization of the respective items in the questionnaire provide additional hints on how “improvement” or “enhancement” can be operationalized by respective solution artefacts.

## 6 Discussion and Outlook

In the preceding chapters, we have shown how problem analysis data can be used to systematically explore a design problem class, identify design factors and design situations, and transform these insights into solution artefact design. Based on the outlined artefact design, generic solutions like reference models and methods can be constructed and subsequently adapted to specific problems. While this procedure is based on survey data of related design problems in organizations, it can be applied to any problem within the addressed design problem class. For every design problem, the “closest” design situation of that class can be identified, and a solution is then based on the respective generic solution artefact.

A survey of IT/business alignment problems has been used to illustrate the proposed approach. While this problem class proved suitable to obtain a sufficiently sized data set which moreover showed very good sampling quality for the applied instruments, only three design factors were identified. As a consequence, the four discovered design situations could only be associated with comparably simple and straightforward design fragments, and the resulting aggregate solution artefact design was much simpler than in other design problem classes. For other design problem classes, we observed six to eight design factors and more complex solution artefacts.

The illustrative survey example also did not exhibit a differentiation between as-is problem data and to-be goal data. If as-is as well as to-be data are collected, different types of situations (as-is vs. to-be) can be discovered in parallel, and much better targeted design activities / project types can be identified.

Another simplification of the used application example is the lack of tradeoffs between design goals and design activities: Since all qualities should be achieved equally, the derivation of “movement” fragments and their aggregation into situational artefacts was straightforward. If tradeoffs have to be observed, both the fragment specification and the fragment aggregation become much more complex.

An interesting feature of many design problem analyses that yield a larger number of design factors is that the first factor is often representing many and quite diverse

problem aspects that are sometimes not easy to interpret semantically. With regard to design problem analysis and solution construction, we interpret this “technically” overloaded factor as a problem independent aggregation of “generalized” properties and the respective solution fragment as a basic set of domain-independent problem solution activities like e.g., general project/transformation management. This aspect of our approach does certainly need additional research effort.

In addition, our approach does not explicitly cover yet the adaptation of situated solution artefacts to specific design problems. On the one hand, we consider this extension not too problematic because there is a plethora of adaptation knowledge on reference models which promises to be generalizable to generic artefacts (including methods). On the other hand, adaptation efforts might depend on problem properties and influence the “optimal” level of artefact genericity that up to now is determined using “technical” homogeneity/heterogeneity metrics only.

Another and probably the most important extension would be to include not only adaptation effort, but also other “economical” properties like the number of problem instantiations of a type or the attractiveness of problems in terms of economic gains into the identification procedure of design factors and in particular design situations.

**Acknowledgements** The IT/business alignment survey has been designed, conducted and interpreted together with Jan Saat, University of St. Gallen, as well as Ulrik Franke, Robert Lagerström and Mathias Ekstedt, all from KTH Stockholm.

The author wishes to thank Christian Fischer and Felix Wortmann, both University of St. Gallen, for fruitful discussions on constructional modelling methodology and for comments on this paper and its predecessor.

## References

- Aier S, Winter R (2009) Virtual decoupling for IT/business alignment—conceptual foundations, architecture design and implementation example. *Bus Inf Syst Eng* 51(2):150–163
- Baskerville RL, Pries-Heje J et al (2009) Soft design science methodolog. In: Proceedings of the 4th international conference on design science research in information systems and technology (DESRIST 2009), ACM, New York
- Baumöl U (2006) Methodenkonstruktion für das Business-IT-Alignment. *Wirtschaftsinformatik* 48(5):314–322
- Becker J, Delfmann P et al (2002) Konfigurative Referenzmodellierung. In: Becker J, Knackstedt R (eds) *Wissensmanagement mit Referenzmodellen. Konzepte für die Anwendungssystem- und Organisationsgestaltung*. Physica, Heidelberg, pp 25–144
- Becker J, Delfmann P et al (2007) Adaptive reference modeling: integrating configurative and generic adaptation techniques for information models. In: Becker J, Delfmann P (eds) *Reference modeling*. Physica, Heidelberg, pp 27–58
- Bucher T, Klesse M (2006) Contextual method engineering. Institute of Information Management, University of St. Gallen, St. Gallen
- Bucher T, Klesse M et al (2007) Situational method engineering—on the differentiation of “context” and “project type”. *Situational method engineering—fundamentals and experiences*. Springer, Boston
- Cattell RB (1966) The scree test for the number of factors. *Multivariate Behavioral Res* 1(1):245–276
- Gammelgård M, Ekstedt M et al (2006) A categorization of benefits from IS/IT investments. In: 13th European conference on information technology evaluation (ECITE 2006), Academic Conferences International, Genoa

- Gustafsson P, Franke U et al (2008) Quantifying IT impacts on organizational structure and business value with extended influence diagrams. In: *The practice of enterprise modeling (PoEM 2008)*. Springer, Stockholm
- Hair JF Jr, Black B et al (2006) *Multivariate data analysis*. Prentice-Hall, Englewood Cliffs
- Härdle W, Simar L (2003) *Applied multivariate statistical analysis*. Springer, Berlin
- Henderson JC, Venkatraman N (1999) Strategic alignment: leveraging information technology for transforming organizations. *IBM Syst J* 38(2/3):472–484
- Hevner AR, March ST et al (2004) Design science in information systems research. *MIS Q* 28(1):75–105
- Johnson P, Ekstedt M (2007) *Enterprise architecture: models and analyses for information systems decision making*. Studentlitteratur, Pozkal
- Kaiser HF (1958) The varimax criterion for analytic rotation in factor analysis. *Psychometrika* 23(3):187–200
- Kaiser HF, Dickman KW (1959) Analytic determination of common factors. *Am Psychol Rep* 14:425–430
- Kaiser HF, Rice J (1974) Little Jiffy, Mark Iv. *Educ Psychol Meas* 34(1):111–117
- Luftman JN (2000) Assessing business-IT alignment maturity. *Commun AIS* 4(14):1–50
- Luftman JN (2005) Key issues for IT executives 2004. *MISQ Executive* 4(2):269–285
- Luftman JN, Kempaiah R (2008) Key issues for IT executives 2007. *MISQ Executive* 7(2):99–112
- Luftman JN, Kempaiah R et al (2006) Key issues for IT executives 2005. *MISQ Executive* 5(2):81–99
- Luftman JN, Kempaiah R et al (2009) Key issues for IT executives 2008. *MISQ Executive* 8(3):151–159
- March ST, Smith GF (1995) Design and natural science research on information technology. *DSS* 15(4):251–266
- Plazaola L, Flores J et al (2007) An approach to associate strategic business-IT alignment assessment to enterprise architecture. In: *5th conference on systems engineering 2007 (CSER2007)*, Stevens Institute of Technology, Hoboken
- Saat J, Winter R, Franke U, Lagerström R, Ekstedt M (2011) Analysis of IT/business alignment situations as a precondition for the design and engineering of situated IT/business alignment solutions. *44th Hawaii international conference on System Sciences (HICSS-44)*, Kauai, HI
- Sledgianowski D, Luftman JN (2005) IT-business strategic alignment maturity: a case study. *J Cases Inf Technol* 7(2):102–120
- Thompson BJ (2004) *Exploratory and confirmatory factor analysis: understanding concepts and applications*. American Psychological Association, Washington
- Winter R (2011) Problem analysis for situational artefact construction in information systems. In: Carugati A, Rossignoli C (eds) *Emerging themes in information systems and organization studies*. Physica, Heidelberg, pp 97–113
- Zachman JA (1987) A framework for information systems architecture. *IBM Syst J* 26(3):276–292