Using Empirical Foundations for Designing EIS Solutions

Robert Winter

Institute of Information Management, University of St. Gallen, Mueller-Friedberg-Str. 8, 9000 St. Gallen, Switzerland
Robert.Winter@unisg.ch

Abstract. From a design science research perspective, enterprise information systems (EIS) are understood as artifacts intended to support organizations in achieving certain goals. Proper EIS design needs not only to be based on solid general foundations ('kernel theories') and valid construction processes, but also should incorporate domain related experience and expertise. One important aspect is to understand which design goals and context factors have lead to which variations in existing solutions in the real world. Another aspect is to understand which design variations can be empirically related to which design goals, and to derive respective design actions. Using examples from enterprise architecture management and process performance management for illustration purposes, we show that existing variations of EIS solutions can be transparently explained and that innovative EIS solutions can be systematically constructed.

1 Introduction

According to Hevner et al. [1] two scientific approaches characterize much of the research in the information systems discipline, namely *behavioral science* and *design science*. Behavioral science addresses research through the *development* and *justification* of theories that explain or predict existing phenomena in a domain. Design science addresses research through the *building* and *evaluation* of innovative artifacts that are intended to solve important, relevant design problems in the domain. Both types of research reflect the respective foundations and methodologies in order to provide guidance for researchers. The knowledge base, as defined in the information systems research framework of Hevner [2] interfaces these approaches. It is composed of foundations and methodologies.

In the wide field of information systems, being concerned with people, task and technology [cf. e.g. 3, 4], different design science research approaches have been proposed over the years. Such approaches aim at constructing and testing various kinds of designed artifacts as solutions to certain classes of design problems in organizations. Since design proposals are often widely varying with regard to foundations, goals, and processes, Hevner [2] introduced a general framework comprising three cycles – relevance, design and rigor cycles –, which should be present and clearly identifiable in every piece of design science research (see Fig. 1).

In the *relevance cycle* the requirements of the problem domain are defined and introduced into the design process. Additionally, the proposed artifacts are established

C. Møller and S. Chaudhry (Eds.): CONFENIS 2011, LNBIP 105, pp. 29–44, 2012.
© IFIP International Federation for Information Processing 2012

in the environment (e.g. by field-testing) in order to demonstrate their problem solving utility. In the *rigor cycle* not only scientific theories and methods as well as existing design products and processes, but also domain experience and expertise are introduced into the design process. Additionally, new generalizable knowledge derived from the design process is added to the knowledge base for reuse. The *design cycle*, which is essentially a solution search procedure, iterates between the core activities of constructing and evaluating design artifacts.

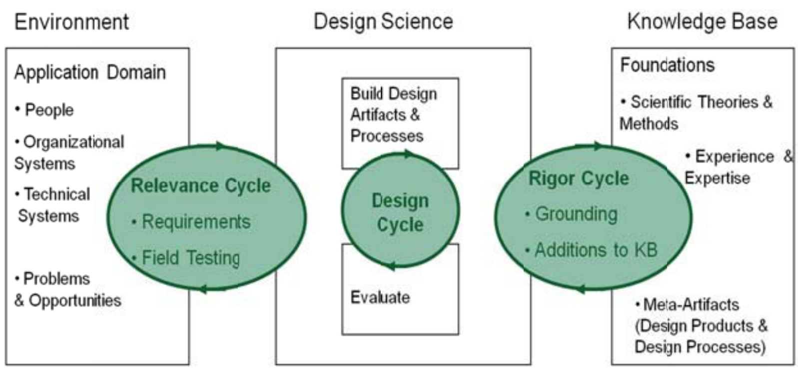

Fig. 1. Design Science Research Cycles [2]

The design and relevance cycles have been subject to many studies and discussions from the early years of design science research on. With more attention being directed to the rigor cycle recently, in particular the role of various types of theories has been investigated [e.g. 5, 6, 7, 8]. Domain related experience and expertise are however often overlooked in the methodological reflection of design science research. The reason might be that, in contrast to theories, methods/processes and other meta-artifacts, domain related knowledge is diverse and not only hard to generalize, but also difficult to obtain and make reusable.

This paper therefore focuses on the role of domain related experience and expertise for design science research in general and for the design of enterprise information systems (EIS) in particular. The goal is to investigate the role and generalizability of domain knowledge for the assessment of existing solutions, the construction of to-be solutions and the transformation process. The relevant foundational concepts are introduced in section 2. For the assessment phase, we present an approach that analyzes existing real-world solutions in a domain to identify possible design goals and relevant context factors for that domain (section 3). For the construction phase, we present an approach that relates design goals in a domain to variations of the respective to-be artifacts and that derives respective design actions (section 4). Examples from enterprise architecture management and process performance management are used in sections 3 and 4 to illustrate that existing variations of EIS solutions can be transparently explained and that innovative EIS solutions can be systematically constructed. The paper is concluded by a discussion and an outlook on further research in this field.

2 Foundational Concepts

Common design artifacts produced by design science researchers in the information systems field are *constructs, models, methods* and *instantiations* [9]. Hevner et al. describe these artifacts as follow: "*Constructs* provide the language in which problems and solutions are defined and communicated […]. *Models* use constructs to represent a real world situation – the design problem and its solution space […]. *Methods* define processes. They provide guidance on how to solve problems, that is, how to search the solution space. […] *Instantiations* show that constructs, models, or methods can be implemented in a working system." [1].

It is important to understand the artifact types of design science research in the information system field not as separate concepts, but as an interdependent system. Winter [10] refers to Chmielewicz's [11] conceptualization of research in social sciences, which may serve as a foundation to explain such dependencies. Chmielewicz differentiates between *ontological facts*, *theoretical statements*, *technological statements*, and *normative statements*. These concepts are represented by the artifact types foundational concepts, cause-effect relations, means-ends relations and justifications (for choosing certain goals or preferring certain means to pursue a goal). Chmielewicz's concept system can therefore be easily matched to design science research artifacts.

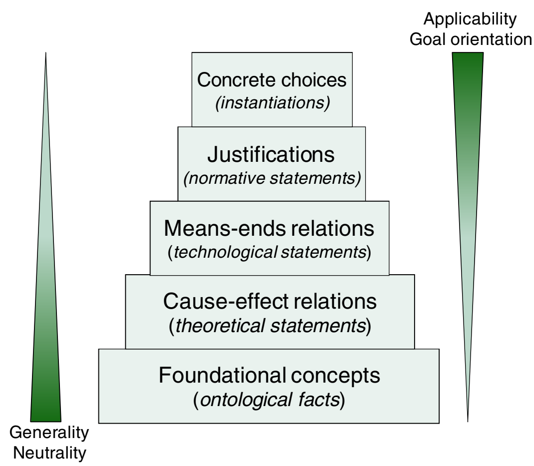

Fig. 2. The interdependent system of concepts for design research

To illustrate the relationships between the concepts and artifacts, the pyramid metaphor can be used: Applicable ontology and meta models define constructs and constitute the foundation for formulating theories for analysis, explanation and/or prediction. Together with domain experience and expertise, valid explanatory and/or predictive theories then constitute the foundation for constructing effective technologies (e.g. reference models or methods). Since alternative means might be effective for the same ends, normative statements are needed to justify which goals to choose and which technologies to apply - Chmielewicz designates this layer "philosophy". At the top of the concept pyramid, concrete goal and technology choices lead to actual design instantiations. Fig. 2 illustrates the concept pyramid.

It is obvious that the presented concepts are fundamentally different regarding the extent they embody certain design goals and regarding their problem-solving power for specific design problems. Goal orientation (vs. neutrality) is strongly related to applicability (vs. generality): Although design artifacts of every type (except instantiations) can be defined on different levels of generality, the typical generality decreases when moving up in the concept hierarchy. In addition to utility, generality is an important quality of design science research artifacts [1]. Baskerville et al. [12] demand a design science research artifact to "represent [.] a general solution to a class of problems."

In addition to applicability / goal orientation vs. generality / neutrality, design artifacts can be classified regarding the problem solution phase they support: While the same system of foundational concepts should be used both in the assessment and construction phases of a problem solution to ensure consistency, instantiations always relates to a specific phase. Certain theories might be used to explain assessment as well as justify design, while others are specific to one of these phases. Since the goals of assessment and construction are different, different technologies will be relevant, and different (yet coherent) justifications will be needed.

3 Assessment: Identifying Contingencies of Existing Solutions

There is some, but not much, work on how to identify contingencies in design science research. For method engineering, "project size", "number of stakeholder groups" or "applied technology" have been suggested as general contingencies [e.g. 13, 14]. Other authors recommend to identify and specify situations individually on a case-by-case basis [e.g. 15]. As a compromise between these positions, we recommend to identify specific contingencies for a domain, i.e. a class of similar design problems, by analyzing existing real-world solutions in that domain. This analysis is not restricted to methods; It is applicable to other problem solutions like (reference) models and constructs as well.

Based on earlier proposals by Winter [16] and Bucher and Klesse [17], the following procedure is proposed in [18] to identify contingencies of existing problem solutions:

Step 1: Preliminary Specification of the Design Problem Class

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 ideas about design goals for the respective class of design problems.

After a while, EIS management practice develops a common understanding about the scope of relevant artifacts and about useful design goals for such problem classes. In the remainder of this section, enterprise architecture management (EAM) is used as an exemplary design problem class. In EAM, a considerable amount of consensus exists regarding which artifacts and relationships should be addressed by that approach. Furthermore, design goals like transparency, consistency, simplification or flexibility are established.

Step 2: Identification of Contingency Factor Candidates

A literature analysis is conducted in order to identify contingency factor candidates for the respective class of design problems, i.e. factors which might have influence on how such design problems are solved in practice.

For EAM, such an analysis yields factors like ‘EAM’s main sponsor is IT or business’, ‘EAM’s main deliverable is maps, analyses or project support’, ‘EAM’s main goal is transparency, consistency, simplification, or flexibility’, or ‘EAM’s role is active or passive’.

Step 3: Field Study

A field study is conducted in order to analyze how design solutions for this class of design problems in practice are actually related to which contingencies. Using principal component analysis on the field study data, the list of potential contingency factor candidates from step 2 is reduced and aggregated into a smaller set of relevant “design factors”. Design factors are usually aggregates of several relevant contingency factors and therefore need to be semantically interpreted.

For EAM, principle component analysis on EAM practice solutions yielded eight design factors (like IT operations support, integrative role, business strategy support, or design impact) which aggregate 54 statistically relevant contingencies. E.g., the design factor ‘integrative role of EAM’ aggregates the contingencies ‘EAM takes place in an interdisciplinary team’, ‘EAM team and business departments continuously exchange information (e.g. in architecture boards)’ and ‘EAM team and IT departments continuously exchange information (e.g. in architecture boards)’.

Step 4: Redefinition of the Design Problem Class

Every surveyed real-world solution in the domain can be understood as a point in a multi-dimensional coordinate system where every dimension corresponds to a design factor. The design problem class now should be redefined by specifying value ranges for the design factors identified in step 3. This means that “outlier” solutions are excluded from further analysis in order to ensure a useful degree of solution homogeneity.

Step 5: Solution Similarity Analysis

Now ultrametric distances can be computed that represent the similarity (or dissimilarity) of the relevant solutions. Metrics are usually based on Euclidian distance. The observations and their distances can be visualized by a dendrogram-like tree graph. The (dis)similarity of two design solutions corresponds to the generality level of their linkage. If two design solutions are very similar, their linkage is represented on a low level of generality. If two design solutions are very different, their linkage is represented on a very high level of generality.

Figure 3 exhibits a tree graph of 119 observed EAM ‘cases’ (vertical axis) in 94 different companies. Their ultrametric distances are represented on the horizontal axis. Case 72 and case 73 are very similar, but differ significantly from cases 6, 12 and 104. The generalization of these five cases (B) is still quite homogeneous compared to the overall, “one size fits all” EAM generalization (A).

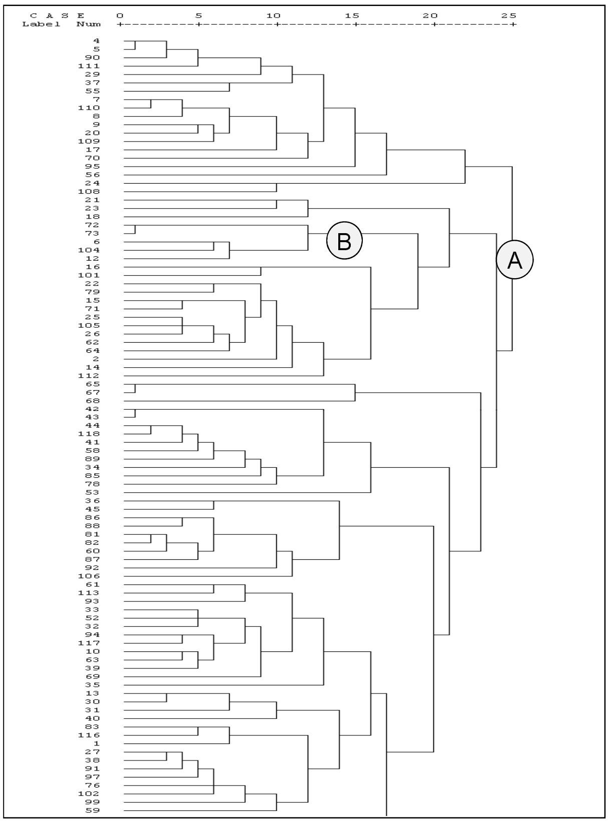

Fig. 3. Visualization of similarities of observed EAM solutions in companies

Step 6: Identification of Representative Design Solutions

In order to not only visualize, but characterize generic design solutions in a domain, a clustering algorithm can be applied to the observation data. By agglomerative clustering, solutions can be specified at any generality level between “full detail” (i.e. one cluster per original observation) and “one size fits all” (i.e. one generic solution description for the entire design problem class). By analyzing the clustering error in relation to the number of clusters, an optimal level of generality (i.e. an optimal number of clusters) can be determined.

For the EAM approaches in 94 companies, the optimal number of clusters is three [19]. This means that, for this observation, three different EAM 'approaches' with specific characteristics should be differentiated (see Fig. 4). With more and broader surveys, this kind of findings might be generalized to the respective design problem domain in general.

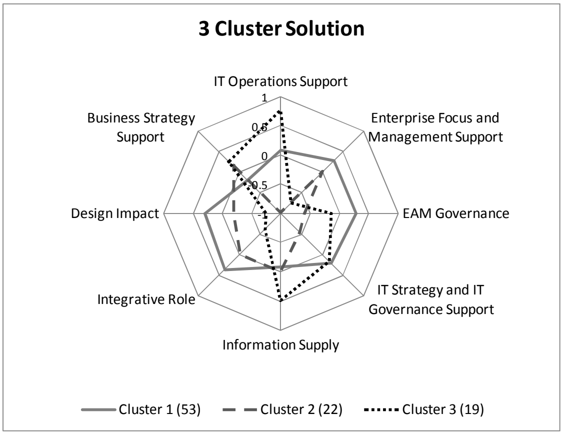

Fig. 4. Problem class decomposition into design situations [19]

Step 7: Specification of Design Situations

For the level of solution description generality chosen in step 6, each cluster represents one design situation. The situations should not only be defined formally (i.e. by specifying value ranges of the respective design factors), but also should be interpreted semantically ("design problem types").

The three EAM clusters differ in particular with regard to their values for the design factors 'IT operations support', 'integrative role', 'design impact', 'enterprise-wide focus' and 'IT strategy support'. These differences are used to characterize one cluster as 'balanced, active EAM', one as 'business analysis' and one as 'IT focused, passive EAM' [19]. In order to interpret the design situations, the cluster centroids have to be analyzed. These can be identified by the mean factor values within each cluster. The mean factor values for each cluster are depicted by the net diagram in fig. 4. The clusters can then be described as follows: [19]

Design situation 1: Balanced, Active EAM

Cluster 1 (solid line in fig. 4) presents a rather balanced approach to EAM. For most factors this cluster shows the highest or at least average values. Especially the similar values for the factors 'IT operations support' and 'enterprise-wide focus' allow the conclusion that organizations within this cluster focus neither on IT support nor on management support. In contrast to the other clusters, the high support of 'IT operations management', 'IT strategy' as well as the focus on 'design impact', 'integrative role' and 'EAM governance' argue for a high degree of integration within the organization. In particular the values for 'design impact', 'integrative role' and 'EAM governance' are by far the highest between all three clusters. It can therefore be presumed that these organizations have a rather high level of maturity in their EAM approach. It should be noted that this cluster includes 53 out of 94 organizations, which lead to the supposition that this cluster represents a 'mainstream' approach.

Design Situation 2: Business Analysis
The second cluster (dashed line in fig. 4) groups 22 organizations that have an apparent focus on business support in their EAM approach. The factors 'IT operations support' as well as 'IT strategy and IT governance support' are clearly assigned with comparatively low values. Comparing the mean factor values to those of cluster 1, the overall low values imply that the organizations in this cluster do not show a high degree of EAM implementation in any dimension. Two conclusions can be derived from this fact: First, the organizations could have decided to apply a minimalist EAM approach, focusing on management support without putting resources in EAM governance or an active role of EAM. Second, the introduction of EAM could only recently be initiated by management and is not very mature yet. For both cases, literature suggests that a sustainable EAM approach can only be established by realizing an effective EAM governance [20, 21].

Design Situation 3: IT Focused, Passive Approach
Organizations assigned to this cluster (dotted line in fig. 4) clearly emphasize the use of EAM for IT operations as well as the information supply by EAM. In contrast, values for 'management support' are by far the lowest compared to the other clusters. As the factors 'design impact' as well as 'integrative role' are not focused by this approach, it can be described as a passive approach that is most probably realized very locally within the organization. Obviously, this small cluster which includes only 19 of the 94 organizations represents a specialized, IT-centered EAM approach that primarily takes a documentation role. It can be presumed that the EAM approach was initiated by IT departments and has not been disseminated throughout the organization yet.

Examples like [22, 23, 24, 25, 26] show that, for typical design problem classes, between four and eight design factors can be identified which explain the variance of the observed design solutions sufficiently. These design factors span up a solution room where between three and six design situations can be differentiated.

4 Design: Relating Design Goals to Solution Variations

While identifying design situations is of utmost importance to understand the design problem class, the construction of innovative, useful solutions as the core phase of design science research has not been covered yet.

In contrast to as-is solutions that exist in the real world and can be analyzed using field studies or other methods in a descriptive way, to-be solutions need to be constructed and have a prescriptive character. It is therefore necessary to base the construction on (1) valid theories, (2) proven design products and processes, or on (3) domain experience and expertise.

- First, explanatory and/or predictive theories can provide important input for solution construction. E.g., indicators from valid success factor models of the domain at hand provide guidance which properties to-be solutions need to have. Such indicators can often be related to specific design activities.
- Second, existing solutions in the domain can be evaluated, and high-performance existing solutions can be chosen as to-be (reference) solutions. This approach however requires that the design problem domain is already quite mature and high-performance solutions exist. When comparing low-performance as-is solutions with high-performance to-be solutions regarding their design factor values, 'elementary movements' can be determined. These 'movements' represent elementary design steps that need to be composed to construct a better solution – or to be more precise: to change existing design solutions in a way that promises to develop properties of a solution that has better performance.
- If high-performance solutions do not exist or cannot be identified, a third way to determine to-be solutions is to use a field study to define desired solution characteristics. If an as-is solution characterization and a to-be characterization are available for all cases in the field study, solution 'paths' can be identified. These paths constitute the project types for situational artifact construction.

Empirical domain knowledge is essential regardless which approach is taken. Since "one size fits all" solutions cannot be expected to perform in all situations, theory application as well as as-is solution ranking and to-be solution surveys need to relate to the identified relevant contingencies, i.e. to be adapted to design situations. For each design situation, there is a different set of design activities (approach 1) or 'elementary movements' (approach 2) or a situation-specific to-be solution (approach 3). In the following, we illustrate approach 2 using the EAM example and approach 3 using an example from Process Performance Management (PPM).

4.1 Solution Composition from Elementary Design Actions (Approach 2)

First, characterizing design factors have to be identified for every design situation. In the EAM example, only situation 'IT focused, passive EAM' is characterized by high values of the design factor 'IT operations support' and low values of the design factors 'enterprise-wide focus', 'integrative role' and 'design impact'. The situation 'balanced, active EAM', in contrast, exhibits much smaller values for 'IT operations support', but much higher values for 'enterprise-wide focus', 'integrative role' and 'design impact'. With regard to 'information supply', 'business support' and 'IT strategy and IT governance support', these two design situations are not much different, so that these factors are not useful to characterize them.

In a second step, characterizing design factors are linked to design problems. For the EAM example, the description of the clusters implies a characterization of the respective design problems:

- In situation 2 (Business analysis), EAM implementation and impact need to be addressed. Business is the main stakeholder and executor and implementation considerations are widely neglected.
- In situation 3 (IT focused, passive approach), EAM is too IT centric and has not disseminated through the entire organization yet. The characterizing design factors ‘integrative role’, ‘enterprise-wide focus’ and ‘design impact’ can be associated with an EAM setup where the main EAM sponsor is the CIO, the main EAM customer is the IT function, EAM is primarily performed within the IT function, and EAM is widely ignored by business units.
- If EAM is not systematically addressed in the organization at all, either a balanced approach (very challenging) or an IT focused EAM or a business analysis approach can be followed.

Most EAM setups can be easily linked to major EAM challenges as often described in the literature. E.g., missing business involvement and missing business value creation of EAM correspond to the first EAM setup, while missing ‘grounding’/‘execution’ and too much ‘locality’ of EAM correspond to the latter EAM setup.

In a third step, elementary design actions are derived by comparing design solutions with design problems. Based on the design situation characteristics of the EAM example illustrated by Fig. 4, the following design actions can be derived:

- In situation 2 (Business analysis), ‘IT operations support’, ‘IT strategy and IT governance support’ as well as ‘EAM governance’ need to be strengthened. Since IT topics and EAM governance constitute widely different measures, two different design actions (designated as A and B) should be differentiated.
- In situation 3 (IT focused, passive approach), ‘design impact’, ‘integrative role’ and ‘enterprise-wide focus’ need to be strengthened. Since design impact and IT/business alignment issues constitute widely different measures, design actions C and D should be differentiated.
- If EAM is not systematically addressed yet and an IT focused approach is favoured, ‘IT operations support’, ‘IT strategy and IT governance support’, ‘information supply’ and ‘business strategy support’ need to be developed foremost. In addition to design action A, a design action E should be defined to address business strategy support and a design action F to address information supply.
- If EAM is not systematically addressed yet and a business analysis approach is favoured, ‘enterprise-wide focus’ as well as ‘information supply’ and ‘business strategy support’ need to be developed foremost. As a consequence, the already defined design actions D, E and F are most relevant.

The final step is to define rules for combining the specified elementary design actions. The fewer characterizing design factors and the fewer design problems have been identified, the simpler the design action configuration will be – and vice versa. For the EAM example, four situated design solutions are configured from, depending on the design situation, up to four design actions out of a total number of six reusable design actions A...F:

- Situated solution I "from business analysis to balanced, active EAM" is comprised of design actions A and B
- Situated solution II "from IT focused, passive to balanced, active EAM" is comprised of design actions C and D
- Situated solution III "initial development of IT focused, passive EAM" is comprised of design actions A, E and F
- Situated solution IV "initial development of business analysis" is comprised of design actions D, E and F

Although not advised because a big maturity leap is necessary, it is possible to combine EAM design methods III and IV to a fifth situated method "direct move to balanced, active EAM" that is composed of design actions A, D, E and F.

4.2 Survey-Based Solution Path Derivation (Approach 3)

Although the above presented approach allows to systematically construct 'better' problem solutions, it (a) requires that such approaches can already be observed and (b) creates no useful solution in cases that already come close to the 'best' solution cluster.

As an alternative, to-be solutions (in the sense of information systems with certain desired properties) could be elicited by means of field studies, i.e. in the same way that is used to identify design situations. In addition to as-is properties of existing solutions, to-be properties of desired solutions are collected and undergo identical analyses.

Using PPM as an example, Fig. 5 illustrates that as-is situations differ significantly from to-be situations. One explanation could be that desired solution properties are

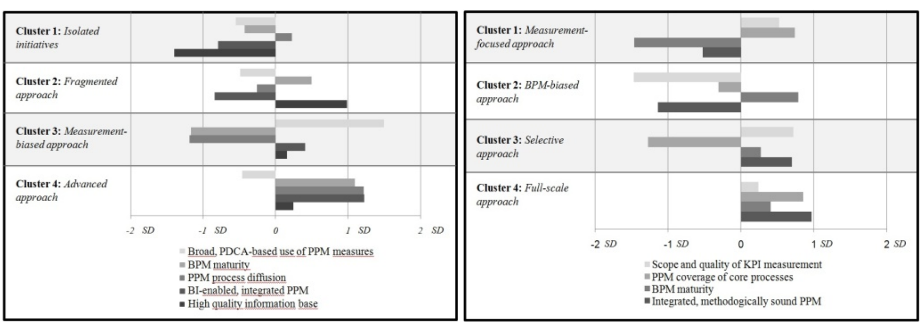

Fig. 5. As-is situations (left) vs. to-be situations (right) for process performance management [27]

more homogeneous than the existing variety of approaches so that cluster analysis generates situations that are less different. This would however require that companies' visions for 'ideal' solutions are very similar which is only true in very mature problem domains. In the majority of domains (including PPM) it can be expected that, if many contingencies exist, companies' PPM visions are quite different so that the variations of to-be solutions is not smaller than the variation of as-is solutions.

From the analysis of as-is and to-be situations, it is not immediately clear which project types result that need to be supported by suitable artifacts. If as-is and to-be data were obtained by the same survey, 'movements' can be analyzed because every case can be assigned to a specific as-is cluster as well as to a specific to-be cluster. Fig. 6 illustrates the aggregated 'movements' for the PPM survey. The number in the lower right corner of each of the boxes indicates the population of each cluster. The thickness of the lines is proportionate to the number of 'movements'. It becomes evident that, although there is a favorite to-be cluster ('full-scale PPM', 23 out of 42 companies), some companies also aim at alternative to-be solutions.

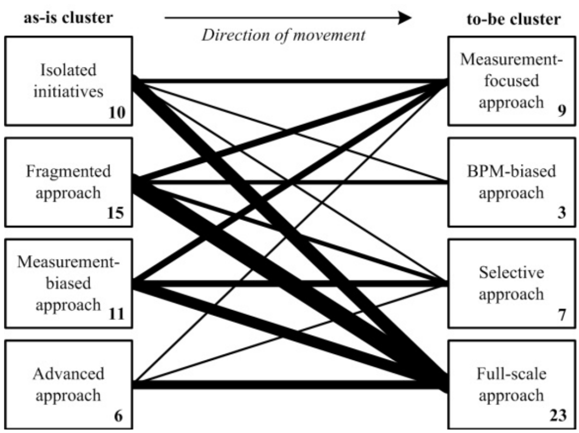

Fig. 6. Project types derived from process performance management design situation analysis [27]

For solution construction, the most significant aggregate 'movements' can be defined as project types. For every project type, it is possible to specify design activities by matching to-be solution properties to as-is solution properties.

To this end a weighted adjacency matrix is used that is based on the items that are grouped by the to-be factors (cf. Table 1). The values represent the arithmetic means of the differences between the to-be and as-is item values across all movers per project type. This allows to identify the areas that require the most change when a company intends to reach a certain to-be PPM approach. Details of this analysis are documented in [27].

Table 1. Adjacency matrix for process performance management [27]

as-is cluster / to-be cluster	Isolated initiatives		Fragmented			Measurement-biased				Advanced	
		delta		delta			delta				delta
to-be factors *Items*	as-is cluster means	Full-Scale	as-is cluster means	Measure-ment-focused	Full-Scale	as-is cluster means	Measure-ment-focused	Selective	Full-Scale	as-is cluster means	Full-Scale
Scope and quality of KPI measurement											
Adherence to schedules is measured for processes.	2.2	1.7	2.6	1.7	1.3	3.5	1.0	1.0	0.8	3.0	1.3
Quality is measured for processes.	1.6	2.2	2.2	2.7	1.4	3.0	0.7	1.3	1.4	2.7	1.5
Process cycle times are measured.	1.8	1.7	2.4	2.0	1.5	2.8	1.3	1.3	1.2	3.4	0.8
Process costs are measured.	2.1	1.8	2.3	2.0	1.3	2.8	0.7	1.3	1.8	2.9	1.3
Capacity utilization is measured for processes.	2.2	1.3	2.8	2.0	1.1	2.8	1.3	1.3	1.2	3.3	1.0
Process resource utilization is measured.	2.2	1.7	2.5	2.0	1.6	3.0	0.7	1.0	1.4	3.1	1.3
Data quality is consistently high.	1.7	2.7	3.4	0.7	1.1	3.1	1.7	1.0	1.4	2.7	1.5
A central integrated data base is in place (e.g., an Enterprise DWH).	1.6	2.3	3.8	1.0	1.0	3.1	1.3	1.3	0.8	3.7	0.8
Processes have defined process officers.	2.9	1.2	3.4	1.0	1.0	3.8	1.3	1.3	1.4	4.0	0.8
Defined BI governance responsibilities and processes are in place.	1.6	2.0	2.3	2.0	1.8	2.8	1.0	1.3	1.4	3.9	0.5
PPM coverage of core processes											
PPM is deployed for production processes.	2.1	1.0	2.2	1.7	1.6	2.5	0.3	0.0	1.0	4.0	0.8
PPM is deployed for sales processes.	2.4	1.0	2.8	1.0	1.1	2.0	0.7	0.0	1.4	3.4	1.3
PPM is deployed for purchasing processes.	2.0	1.2	1.9	1.7	1.3	1.6	0.7	0.0	1.8	2.9	1.3
PPM also covers non-financial measures.	2.4	1.0	1.8	1.7	0.9	2.1	1.0	0.3	0.6	3.1	-0.3
BPM maturity											
Process orientation is a central paradigm.	2.9	1.5	3.1	1.0	1.5	3.1	1.0	0.7	1.6	3.6	0.8
Process flows are consistent and transparent beyond functional borders.	2.4	1.3	2.9	0.7	1.8	2.3	1.7	1.7	2.4	3.4	1.0
Processes are consistently documented and/or modelled.	2.2	2.2	2.9	1.3	1.5	2.6	1.7	2.0	2.4	3.6	1.0
Process flows are consistent and transparent beyond system borders.	2.3	1.5	2.8	1.0	1.6	2.1	1.0	1.3	3.0	3.1	1.3
Integrated, methodologically sound PPM											
PPM is part of the enterprise-wide Balanced Score Card (BSC).	1.7	2.0	1.4	0.3	1.5	2.5	0.0	0.3	1.0	3.3	1.3
PPM is part of the Corporate Performance Management (CPM).	1.7	2.0	1.8	0.0	2.0	2.4	0.0	0.3	1.0	3.4	1.0
The plan-do-check-act (PDCA) cycle is applied for PPM.	1.4	1.7	2.0	0.0	2.1	2.7	0.0	0.3	0.8	3.3	1.3

In the PPM example, the most important project type (and the first movement described in the adjacency matrix) is the shift from the “isolated initiatives” approach towards a “full-scale PPM” implementation. Organizations running isolated PPM initiatives are especially characterized by the lack of a high quality information base. Moreover, neither from a business intelligence nor from a business process management perspective have these companies established concepts in place. A more detailed analysis reveals that, in order to realize an effectual PPM concept, these companies explicitly focus (a) on providing an adequate IT infrastructure and (b) on gaining a deeper knowledge about their work by consistently documenting their processes. Besides, in order to bundle thus far isolated initiatives, organizations consider integrating their PPM ambitions with an enterprise balanced scorecard and/or their enterprise performance management program. Organizations starting off from the “fragmented approach” strive for either the “measurement-focused” or the “full-scale” approach. As described above the “measurement-focused” approach emphasizes the use of a broad set of key performance indicators (KPIs) for high quality decision making regarding the improvement of core processes—assumedly often in an on-demand setting based on real-time data. While organizations using the “fragmented approach” show a profound process orientation and possess a high quality information base, the transition to the “measurement-focused approach” requires the development of a set of crucial process performance metrics like process quality, cycle times as well as resource and capacity utilization. Moreover, the change demands a stronger concentration on the measurement of core processes in particular on the basis of non-financial metrics. A shift from the “fragmented” to the “full-scale approach” on the contrary, first and foremost necessitates a more organized and well-planned method. In order to overcome the parallelism of a strong process orientation and an underutilized IT infrastructure, organizations rigorously implement the plan-do-check-act cycle and align their process performance initiatives with enterprise performance management. Bridging the business-IT-divide is further enhanced by making process flows transparent across functional borders for the whole workforce. The “measurement-biased approach”—as the name betrays—is characterized by a strong overestimation of KPI use and measurement. Consequently, each convergence towards another approach—be it “measurement-focused”, “selective”, or “full-scale” —requires establishing a more sophisticated process orientation. In any case the measurement-biased organization must consistently design and model its processes and make them transparent beyond both functional and system borders so as to enable a comprehensive and successful PPM implementation. The last movement Cleven et al. [27] analyzed in more detail leads from the “advanced” to the “full-scale approach”. Due to the fact that the advanced approach already shows a comparatively high maturity, only minor adoptions are required. The analysis reveals that an improvement of the overall data quality becomes necessary. As the measurement level of production processes is already very high, the focus will be shifted to sales and purchasing processes. An increased use of the balanced scorecard and the plan-do-check-act cycle further fosters the holism of the approach.

5 Discussion and Future Research

We presented a generally applicable approach to systematically analyze the mutability of information system solutions and differentiate design situations for a domain. For relating design goals to solution variations, we presented two alternative approaches. While the first approach requires an evaluation of the as-is clusters and then helps to transform inferior solutions into superior ones by specifying respective design actions, the second approach is based on a separate set of as-is and to-be situation specifications which provide the foundation to define project types and then derive appropriate design activities. While the latter alternative provides a sufficient variation of to-be solutions for every domain, it requires surveyed organizations to specify their respective solution vision in great detail – which is not always possible in particular when a domain is immature.

While many solution engineering approaches claim to incorporate situational factors, they do nearly never detail what these situational factors exactly are and how they can be incorporated into solution design and solution component configuration rules. This is the contribution of the empirical approach we presented here. It must however be conceded that every empirical foundation of solution design is only as general as the empirical base supports. Within a specific domain like EAM or PPM, we can reliably specify design factors, design situations, design actions and solution procedures. The less specific the domain is, the less useful the results, and the less effective their application will be. Several broad categories of research opportunities exist:

- First, the proposal needs to be further validated. Although a number of research projects have applied the proposed procedures and many case studies provide evidence that as-is situations, to-be situations, project types and respective design actions are meaningful and realistic in the respective domain, these claims might be invalidated by future studies.
- Both the EAM example and the PPM example comprised several simplifications like the elimination of tradeoffs between design goals and design activities. E.g., the derivation of design actions was straightforward for EAM if implementation

impact and business involvement can be achieved equally. If however tradeoffs have to be observed, solution design becomes much more complex.

- An interesting feature of many design solution 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 qualitatively. With regard to design solution analysis and solution construction, we interpret this "technically" overloaded design factor as a problem independent aggregation of "generalized" properties and the respective solution component 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 attention.
- In addition, our approach does not explicitly cover yet the adaptation of situated solutions to specific design problems in an organisation. 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 transferrable to this approach. On the other hand, adaptation efforts might depend on problem properties and influence the 'optimal' level of solution generality that up to now is determined using 'technical' homogeneity/heterogeneity metrics in cluster analysis only.
- Another and probably the most important extension would be to include not only adaptation effort, but also other 'economic' properties like the number of problem instantiations of a type or the attractiveness of design problems in terms of economic gains into the identification procedure of design factors and in particular design situations. This is probably the most interesting – and challenging – avenue for further research.

References

1. Hevner, A.R., March, S.T., Park, J., Ram, S.: Design Science in Information Systems Research. MIS Quarterly 28(1), 75–105 (2004)
2. Hevner, A.R.: A Three Cycle View of Design Science Research. Scandinavian Journal of Information Systems 19(2), 87–92 (2007)
3. Mason, R.O., Mitroff, I.I.: A Program for Research on Management Information Systems. Management Science 19(5), 475–487 (1973)
4. Keen, P.G.W.: Information systems and organizational change. Communications of the ACM 24(1), 24–33 (1981)
5. Gregor, S.: The Nature of Theory in Information Systems. MIS Quarterly 30(3), 611–642 (2006)
6. Gregor, S., Jones, D.: The Anatomy of a Design Theory. Journal of The Association For Information Systems 8(5), 312–335 (2007)
7. Fischer, C., Winter, R., Wortmann, F.: Design Theory. Business and Information Systems Engineering 52(6), 383–386 (2010)
8. Venable, J.: The Role of Theory and Theorising in Design Science Research. In: Proceedings of the 1st International Conference on Design Science in Information Systems and Technology (DESRIST 2006). Claremont Graduate University, Claremont (2006)
9. March, S.T., Smith, G.F.: Design and Natural Science Research on Information Technology. Decision Support Systems 15(4), 251–266 (1995)

10. Winter, R.: Design Science Research in Europe. European Journal of Information Systems 17(5), 470–475 (2008)
11. Chmielewicz, K.: Forschungskonzeptionen der Wirtschaftswissenschaften. Poeschel, Stuttgart (1994)
12. Baskerville, R.L., Pries-Heje, J., Venable, J.: Soft design science methodology. In: Proceedings of the 4th International Conference on Design Science Research in Information Systems and Technology (DESRIST 2009), Philadelphia (2009)
13. Kornyshova, E., Deneckère, R., Salinesi, C.: Method Chunks Selection by Multicriteria Techniques: an Extension of the Assembly-based Approach. In: Situatinal Method Engineering Fundamentals and Experiences, pp. 64–78. Springer, Geneva (2007)
14. van Slooten, K., Hodes, B.: Characterizing IS Development Projects. In: Method Engineering - Principles of Method Construction and Tool Support, pp. 29–44. Chapman & Hall, Atlanta (1996)
15. Mirbel, I., Ralyté, J.: Situational method engineering: combining assembly-based and roadmap-driven approaches. Requirements Engineering 11(1), 58–78 (2006)
16. Winter, R.: Problem Analysis for Situational Artefact Construction in Information Systems. In: Carugati, A., Rossignoli, C. (eds.) Emerging Themes in Information Systems and Organization Studies, pp. 97–113. Physica, Heidelberg (2011)
17. Bucher, T., Klesse, M.: Contextual Method Engineering. Working Paper, Institute of Information Management, University of St. Gallen (2006)
18. Winter, R.: Design Solution Analysis for the Construction of Situational Design Methods. In: Ralyté, J., Mirbel, I., Deneckère, R. (eds.) ME 2011. IFIP AICT, vol. 351, pp. 19–33. Springer, Heidelberg (2011)
19. Aier, S., Gleichauf, B., Winter, R.: Understanding Enterprise Architecture Management Design – An Empirical Analysis. In: Proceedings of the 10th International Conference on Wirtschaftsinformatik WI 2011, Zurich, pp. 645–654 (2011)
20. Aziz, S., Obitz, T., Modi, R., Sarkar, S.: Enterprise Architecture: A Governance Framework - Part I: Embedding Architecture into the Organization (2005)
21. Aziz, S., Obitz, T., Modi, R., Sarkar, S.: Enterprise Architecture: A Governance Framework - Part II: Making Enterprise Architecture Work within the Organization (2006)
22. Baumöl, U.: Strategic Agility through Situational Method Construction. In: Proceedings of the European Academy of Management Annual Conference 2005, München (2005)
23. Bucher, T., Winter, R.: Taxonomy of Business Process Management Approaches: An Empirical Foundation for the Engineering of Situational Methods to Support BPM. In: vom Brocke, J., Rosemann, M. (eds.) Handbook on Business Process Management, vol. 2, pp. 93–114. Springer, Heidelberg (2010)
24. Klesse, M., Winter, R.: Organizational Forms of Data Warehousing: An Explorative Analysis. In: Proceedings of the 40th Hawaii International Conference on System Sciences (HICSS-40). IEEE Computer Society (2007)
25. Lahrmann, G., Stroh, F.: Towards a Classification of Information Logistics Scenarios – An Exploratory Analysis. In: Proceedings of the 42nd Hawaii International Conference on System Sciences (HICSS-42). IEEE Computer Society (2009)
26. Leist, S.: Methoden zur Unternehmensmodellierung - Vergleich, Anwendungen und Diskussionen der Integrationspotenziale. Habilitation, Institut für Wirtschaftsinformatik, Universität St. Gallen (2004)
27. Cleven, A., Winter, R., Wortmann, F.: Process Performance Management – Illuminating Design Issues through a Systematic Problem Analysis. In: The 26th Annual ACM Symposium on Applied Computing, pp. 280–286. Association for Computing Machinery (ACM), Taichung (2011)