PROCEDURALLY TRANSPARENT DESIGN SCIENCE RESEARCH: A DESIGN PROCESS MODEL

Completed Research Paper

Rob Gleasure
Department of Accounting, Finance and Information Systems
University College Cork
Cork, Ireland
robgleasure@gmail.com

Joseph Feller
Department of Accounting, Finance and Information Systems
University College Cork
Cork, Ireland
jfeller@afis.ucc.ie

Brian O'Flaherty
Department of Accounting, Finance and Information Systems
University College Cork
Cork, Ireland
boflaherty@afis.ucc.ie

Abstract

This paper presents a process model for Information Systems (IS) scholars conducting Design Science Research (DSR). The paper argues that the impact of DSR on the IS discipline has been inhibited by perceived limitations in the rigor of DSR studies, and that such perceptions are often due to a lack of procedural transparency surrounding the development process in DSR studies. To help address this issue, the paper presents the Process Model for increased Procedural Transparency (PMPT). A range of design literature is analyzed to identify and characterize the distinct stages of design-oriented research. These stages are laid out in a process model, which is then bound in a recursive instantiation, and demonstrated through an application. The paper concludes with a discussion of the contribution of PMPT, as well as opportunities for future research.

Keywords: Design Science, Design Theory, Process modeling, Transparency

Introduction

Simon (1996, p.111) posited that "schools of engineering, as well as schools of architecture, business, education, law, and medicine, are all centrally concerned with the process of design," and design-oriented research has emerged as an important component of the Information Systems (IS) research landscape, both in the form of 'design theories' (Gregor and Jones 2007; Walls et al. 2004) and 'design science research' (Hevner 2007; Winter 2008). Gregor and Jones (2007) credit design-oriented research with a number of breakthroughs in IS, including structured systems analysis and the systems development lifecycle, commenting that "It is difficult to over-emphasize the significance of design work and design knowledge in Information Systems (IS) for both research and practice" (p.313). However, the relatively low volume and impact of design-oriented research produced by IS scholars in the past does not reflect this perceived centrality of design for the discipline, as many scholars have observed (e.g. Fallman 2003; Hevner 2007; Preece 2002; Walls et al. 2004; Zimmerman 2007).

We argue, in agreement with others (e.g. Iivari 2007; Peffers et al. 2008), that this historic under-appreciation of design-oriented research in IS is due largely to a perceived lack of methodological rigor surrounding the development and evaluation of the "IT Artifact," processes central to DSR studies. The objective of this study is therefore to *develop a procedurally transparent process model for design-oriented research in IS, in order to facilitate demonstrably rigorous design-oriented research*. Specifically, the process model described in this paper represents a Design Science Research (DSR) process, as DSR is arguably the dominant design-oriented paradigm in IS (Walls et al. 2004; Winter 2008).

The next section of the paper examines design-oriented research in IS and in particular the role of *procedural transparency* in enhancing the perceived rigor of studies. Following this, eight key papers on design-oriented research in IS are analyzed, in order to identify and characterize key design components of design-oriented research. We then build upon these design components to present a DSR process model (the Process Model for increased Procedural Transparency (PMPT)). The process model is bound in a recursive instantiation (i.e. it also describes the process of its own development), and demonstrated through an application to extant research. Finally, we present the conclusions of our work, and call for future research to expand the process model to encompass other aspects of methodological rigor.

Design-oriented research in IS

Design-oriented approaches differ from behavioral research, as they focus upon 'artificial' objects and phenomena, which are man-made rather than occurring naturally (Simon 1996). These artificial artifacts represent complex developments in the context of IS (Benbasat and Zmud 2003), hence a number of varying accounts of design-oriented research have been presented within the discipline. These include but are not limited to 'design science' (March and Smith 1995), 'systems development research' (Nunamaker et al. 1990), 'design theory' (Gregor and Jones 2007), and Information Systems Design Theories (ISDTs) (Walls et al. 1992; 2004). Each account presents an alternative perspective on (1) what constitutes an appropriate design artifact, and (2) how to go about developing and evaluating such an artifact.

This dual focus emerges because 'design' refers to both a design *product* and also a design *process* (March and Smith 1995, Hevner et al. 2004). Yet despite the importance placed upon the design process, "methodology research in design science is still struggling to amalgamate findings from engineering (which is even denied 'science' status by some), computer science, management, and also from social sciences into a consistent methodology" (Winter 2008, p.470). Some scholars claim the goal of increased rigor in the design process of design-oriented research is inappropriate, e.g. Cross (2001) argued that "method may be vital to the practice of science (where it validates the results), but not to the practice of design (where results do not have to be repeatable, and, in most cases, must *not* be repeated, or copied)"(p.51). However Iivari (2007) claimed that such a laissez-faire perspective on rigor was detrimental for the advancement of DSR, stating, "*Constructive research methods should make the process of building IT meta-artifacts disciplined, rigorous and transparent… It is this methodological rigor that distinguishes Information Systems as design science from the practice of building of IT artifacts*" (p. 26).

Hevner (2007) presented his support for these claims, stating that "*it is vital that we as a research community provide clear and consistent definitions, ontologies, boundaries, guidelines, and deliverables for the design and execution of high quality design science research projects*" (p.47). More recently, Carlsson et al. (2011) also drew attention to the requirements for higher standards of rigor in design-oriented research, arguing *"as there is not necessarily one way of logically getting from an extant theory to a design theory, some creativity is required... However, a methodological consideration is how to use extant theory and previous knowledge in a rigorous and practical way. Although the potential of increasing relevance of IS research is one of the drivers behind IS design research, it should not be at the expense of rigor"* (p.114).

Although the issue of rigor is complex and expectations are heterogeneous according to the forms of data gathering and analysis employed, one recurring factor in determining the perceived rigor of a study is that of *procedural transparency* (Benbasat et al. 1984; Klein and Myers 1999; Yin 2008). This does not imply that other factors are not relevant. On the contrary, methodological issues such as case study design (Dubé and Paré 2003), or the reliability of survey measurements (Boudreau et al. 2001) are often key to the credibility of research. However, a reader's perceived rigor is likely to suffer if the considerations that led to the resolution of such methodological issues go undocumented. Thus, a significant need is evident in the DSR community for design process models facilitating *procedurally transparent* design processes.

It is noteworthy that Peffers et al. (2008) drew upon design literature from a variety of disciplines adjacent to IS, in particular computer science and engineering, and isolated six common themes in the design process. This was used to create a nominal model for DSR, which the authors called the Design Science Research Model (DSRM). The DSRM provides a strong starting point and facilitates a thorough awareness of the fundamental issues encountered in design-oriented research across multiple academic disciplines. However, the DSRM was designed to act as a means of standardizing how DSR is communicated and evaluated, in a way that is as consistent as possible with previous design-oriented literature. Thus, Peffers et al. focused upon the development of a standard mental model with which to view DSR, in order to provide a shared vocabulary across studies. While this consistency is useful as a means of normalization and comparison, Peffers et al. did not explicitly focus on addressing the issue of *procedural transparency*. This lack of focus on transparency is demonstrated by the minimal attention paid in DSRM to a number of traditionally well-established design components, for example kernel knowledge (c.f. Walls et al. 1992). The documentation of design components such as kernel knowledge is not necessarily part of the legacy of design-oriented literature as "explanations of how and why solutions work, while important, aren't the core issue" (Venable 2006, p.11). As a result, from the perspective of consistency and normalization it was sensible for Peffers et al. (2008) not to describe kernel knowledge explicitly, as much of the design literature does not address this issue. However, the process model developed in this study seeks to address *procedural transparency* directly. This means that an account of such formative design components is of significant value, as it adds detail to an important step in the design process. The diverging focus of this study and that of Peffers et al. (2008) manifests an alternative process model for DSR, which is presented in Figure 1. The next section describes both the components of the model and the methods and process used in the development of this model.

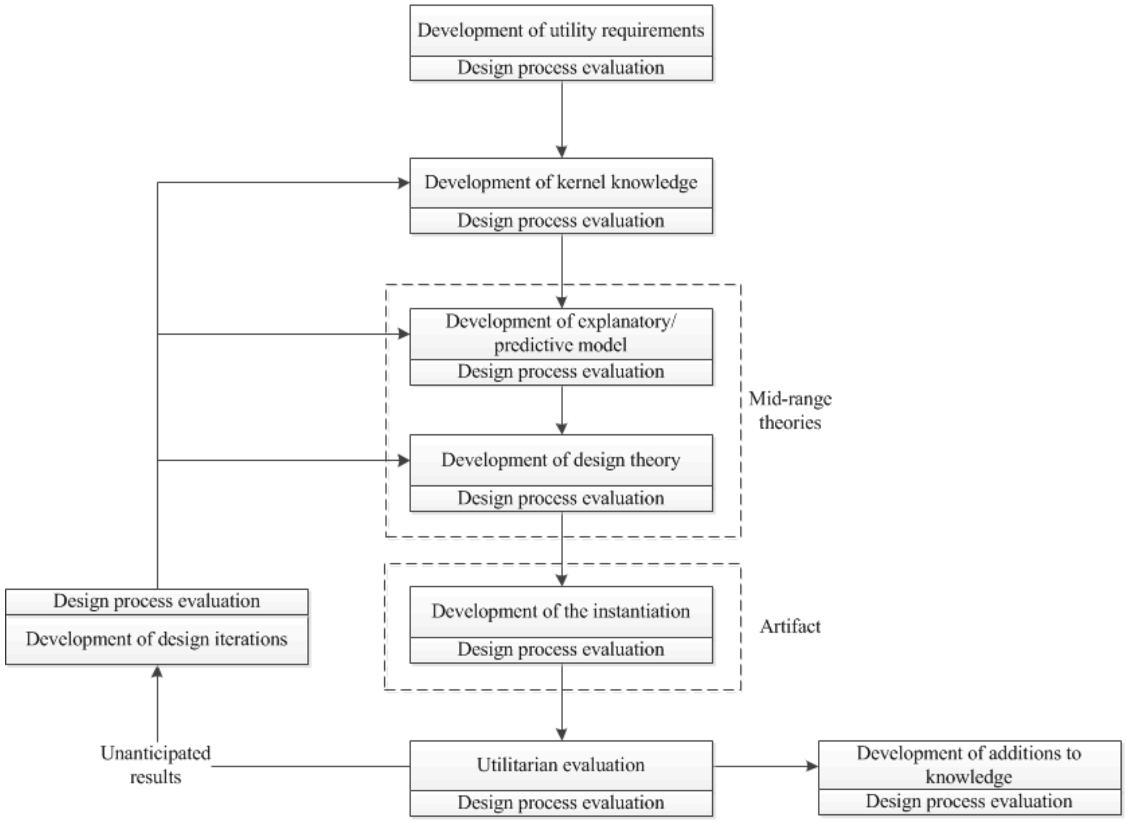

Figure 1. A Process Model for Procedurally Transparent design science research

Developing and Implementing the Process Model for DSR

This section analyzes existing design-oriented IS research, in order to build a complete description of the various stages involved in DSR. The intention of this analysis is similar to the analysis performed by Peffers et al. (2008), in that the desired outcome is a generic abstraction of the design process. Therefore, to build upon progress already made by Peffers et al., this study employs the same literature sources as a starting point, as well as the same basic theory-building approach of inducing a single design process from these sources via a process of abstraction and comparison. However, a crucial difference exists between this study and that of Peffers et al. for both the selection of design-oriented literature sources and the approach to theory building. Peffers et al. sought to generate an abstraction of the design process to identify *commonly documented* significant aspects of design, whereas this study also seeks to identify significant aspects *not commonly documented*. Thus some adaptations to the approach are necessary, as described in the following sections. It is noteworthy that these sections describing the development process also reflect the stages described by the process model. This is done to ensure both the internal consistency of the logical argument presented, as well as the *procedural transparency* of development.

The development of utility requirements

Design-oriented research is often applied to research problems existing within poorly defined and dynamic environments, characterized by complex interactions between technical components and human behaviors (March and Smith 1995). In such areas traditional descriptive research approaches struggle to validate theoretical models, as the isolation and measurement of variables directly is challenging. Like traditional descriptive approaches, design-oriented approaches also possess an underlying theoretical model, which represents how the design will work (Goldkuhl 2004; Kuechler and Vaishnavi 2012). However, design-oriented research is validated by the provision of 'utility', which brings about the desired change in the system (Simon 1996). For example, some conceptualization of certain complex cognitive or social behaviors may exist in the academic literature, but may be difficult to validate due to the complexity of their surroundings. However, if design principles that exploit this conceptualization of complex cognitive behaviors demonstrate utility, then this observed utility provides empirical validation for the underlying understanding. In this way, design-oriented research not only offers useful design prescriptions, it also offers an indirect means of measuring phenomena in complex and poorly understood environments, i.e. what Hevner et al. (2004) call 'wicked problems'. An example of these *utility requirements* is observed in Hanseth and Lyytinen (2010), who developed design principles and rules for information architectures to maintain both simplicity and usefulness, as well as a modular generative adaptability. While the design principles and rules focused upon these two dimensions, the prescriptive quality of the development was ultimately measured in terms of its impact on the success of information architectures. When the design principles achieved the desired utility, the theoretical assumptions behind the principles (the need for simplicity, usefulness and a modular generative adaptability), were indirectly validated.

These *utility requirements* represent the desired change in the problem system and describe the motivation for these changes. Congruent descriptions of this component are presented in each of the sources as 'utility' (Hevner et al. 2004), 'meta-requirements' (Walls et al. 1992), 'purpose and scope' (Gregor and Jones 2007), 'problem identification and motivation' (Peffers et al. 2008), 'awareness of problem' (Kuechler and Vaishnavi 2008), 'general requirements' (Baskerville and Pries-Heje 2010), as well as being captured under the headings of 'conceptual framework' (Nunamaker et al. 1990) and 'problem identification' (Cole et al. 2005), and 'outcome' (Carlsson et al. 2011). In line with this, the *utility requirements* for this study demand a detailed and IS-specific prescriptive process model for the DSR design process, which may subsequently be applied to increase the *procedural transparency* of the design process of DSR studies, as well as bring to light key considerations during each stage of development.

The development of kernel knowledge

The second design component relates to existing academic and industrial knowledge that describes related phenomena. This serves to inform the *utility requirements*, *explanatory/predictive model*, and the actual design prescriptions manifested in the *design theory*. Simon (1996) describes such literature as 'micro theories'. These theories often emerge from the IS academic literature or literature assimilated from adjacent academic disciplines (Gregor and Jones 2007; Hevner et al. 2004; Walls et al. 1992). However, it may also include industrial theory-in-use, rather than solely academic findings (Markus et al. 2002, Sarker and Lee 2002). Adipat et al. (2011) illustrated the role of kernel knowledge in DSR when they proposed an approach to adapting webpage presentation for mobile devices, based on tree-view layouts, hierarchical text summarization, and colored keyword highlighting. Adipat et al. drew upon on a combination of cognitive fit theory and information foraging theory to demonstrate differences in browsing behavior on mobile devices to standard browsers, and consequently why traditional layouts were inappropriate. Adipat et al. emphasized the compatibility of both cognitive fit theory and information foraging theory in the context of mobile interface design. Such a step is important, as any theory being adapted for IS research must not only be of a high quality in isolation, but also a good 'fit' (Weber 2003). Truex (2006) describes this 'fit' as both a reflection of the coverage of theories, the theories' 'intellectual trajectory', and any implications the theories may have for research methods. Each of these three concerns is equally relevant to design-oriented research.

Firstly, an inappropriate theoretical coverage of a problem not only makes design prescription less likely to achieve their required utility, it also makes it more difficult to accurately interpret findings relating to utility. For example, Walls et al.'s development of a design theory for vigilant information systems assumed that executives viewed strategic decision-making as "a process of attention to issues with varying and shifting priorities" (1992, p.49). Thus, Walls et al. focused upon 'triggers' as a means of tracking issues, so improving executive awareness of issues and improving strategic decision-making. Had Walls et al. not been aware of executives' limited awareness of issues and instead targeted design at executives' issue handling directly, a lack of impact on executive decision-making may have resulted. This lack of impact could have been incorrectly attributed to the quality of the design prescriptions, rather than the understanding of the system.

Secondly, a theory developed on an unsuitable intellectual trajectory may have explained phenomena in a way that is not harmonious with the *utility requirements* of the study in question, hence may have some shortcomings that have not been documented. Goldkuhl (2004, p.61) described this as 'internal grounding,' which "means an investigation of internal warrants (as e.g. values and categories) and internal cohesion of the knowledge". For example, in a survey of theories of human vision in cognitive neuroscience, Frintropp (2010) showed that some theories were developed to simulate human vision, whereas others were designed to be implemented in efficient computer vision or robotics systems. Were an IS scholar to develop software interface design prescriptions to enable usability based on the latter category, it is quite possible that their representation of human visual behavior would be inappropriate, thus produce ineffectual design prescriptions.

Thirdly, epistemological and methodological differences in the theories and the design-oriented research may be relevant, i.e. "The grounding of practical knowledge must be done in accordance with the epistemological character of such knowledge" (Goldkuhl 2004, p.63). At a methodological level, Yin (2008) described the ability of qualitative research to provide similar levels of analytical generalizability as quantitative studies, but not similar levels of statistical generalizability. This means that some theories may be more contextually interwoven with their findings than others and are consequently not as readily applied outside of these contexts. This may be relevant where the context intended for design prescriptions differs fundamentally from the context of the theories. For instance, Harrison and Farn (1990) demonstrated that some managerial factors are not as relevant in China as in North America. Thus, if design prescriptions intended for a North America system are developed on the basis of theories induced from case studies in China, a lack of utility could result and again be incorrectly attributed to the quality of the design prescriptions, rather than the understanding of the system.

Walls et al. (1992) argued that synthesized methodology and topic-related literature should be viewed as separate design components, referred to as 'design process kernel theories' and 'design product kernel theories', respectively. According to this distinction, 'design process kernel theories' differ from those relating to the product as they govern the overall design process, so comprise mainly methodology studies. For example, Siponen et al. (2006) developed a design theory for secure information systems according to the framework laid out by Walls et al (1992; 2004). In this instance, Siponen et al. (2006) utilized earlier work on secure information systems as design product kernel theory, and Walls et al. (1992; 2004) as design process kernel theory. The earlier work on secure information systems informed the *utility requirements*, *model and constructs*, and *design theory*, which conceptualized systems in terms of objects, threats and security features. However, the development process, as well as the structure of the design theory, was guided by Walls et al. (1992; 2004).

Kuechler and Vaishnavi (2008) also acknowledged Walls et al.'s distinction when describing the development of a design theory, however they opted to leave their account of this design component blank after they found this failed to describe any 'novel and meaningful' aspects of their study. Perhaps for this reason, Kuechler and Vaishnavi (2008) do not explicitly reference 'design process kernel theory' in their framework, instead referring to collective 'kernel theory'. Both Gregor and Jones (2007) and Baskerville and Pries-Heje (2010) acknowledge and explicitly reject Walls et al.'s (1992) separation of kernel knowledge for design process and those for the design product. They argue that a dualist perspective is not necessary, as both types of theory perform the same function in a study, albeit at different layers in the theoretical foundation. This implies that design prescriptions are ultimately informed by both methodological theories for structure, and other topic-centric theories for semantic content. This 'singular' view of kernel knowledge is referred to under the collective handle of 'justificatory knowledge' by Gregor and Jones (2007), 'capabilitiesVaishnavi' by Baskerville and Pries-Heje (2010), 'applicable knowledge' by Hevner et al. (2004), and 'extant knowledge, data and theories' by Carlsson et al. (2011). Interestingly, Peffers et al. (2008), Nunamaker et al. (1990) and Cole et al. (2010) do not include this design component in their high-level frameworks, although it is implied during their discussion.

The authors of this study concur with the singular view of kernel knowledge, as the separatist view has the potential to become misleading under certain conditions. This is because, as already noted existing academic findings assimilated as product-related *kernel knowledge* are already bound within the context of some methodological assumptions (c.f. Truex 2006), hence literature impacting on the design product also impacts upon the design process. This is particularly acute in this study, in which the *kernel knowledge* for the DSR design process includes design-oriented literature that simultaneously informs both the design process and the process model that represents the design product. To complicate matters further, as the process model presented in this study is presented in a recursive instantiation, i.e. it also describes the process of its own development; it would be impossible to place any of the formative literature in one category and not the other. Thus the singular view of *kernel knowledge* is preferred here to avoid forcing applicants of the design process model to disentangling relevant existing knowledge in a way that may be potentially arbitrary.

With regard to the selection of primary literature sources from which to develop the process model, this study refines Peffers et al.'s (2008) sample literature to be more IS-specific, in order to achieve a finer level of detail and compare descriptions of the design process at a more atomic level. As a result Archer (1984), Takeda et al. (1990), and Eekels and Roozenberg (1991) are removed due to their being exogenous to IS. Two additional sources describing themselves as 'design theory' (Baskerville and Pries-Heje 2010; Gregor and Jones 2007) and two sources describing themselves as DSR (Carlsson et al. 2011; Kuechler and Vaishnavi 2008) are included. These sources were chosen for a number of reasons. Firstly, they are all relatively recent, with only Gregor and Jones (2007) predating Peffers et al. (2008). This avant-gardism of the sources is significant for a methodological approach such as DSR that is still maturing, as it facilitates the encapsulation of recent conceptual breakthroughs. Secondly, each of these sources differs significantly in how they conceptualize design-oriented research. The diversity of the sources affords the identification of asymmetries in the accounts of design. These asymmetries are key to the increase in transparency of a unified process model, as they demonstrate areas in which one account is explicit and another is not. Thirdly, each of these sources builds upon varying foundational literature to the remaining sources. This increases the theoretical coverage by maximizing the indirect inclusivity of literature. For example, March and Smith's (1995) conceptualization of DSR informed much of the work of Hevner et al. (2004), hence March and Smith' perspective is represented to some degree by Hevner et al., albeit at an additional level of separation. Similarly, much of Venable's (2006) and Goldkuhl's (2004) arguments are encapsulated by Kuechler and Vaishnavi (2008), and much of Van Aken's (2005) perspective is represented by Carlsson et al (2011).

The development of the explanatory/predictive model

While the *utility requirements* describe the required change in the problem system, the *explanatory/predictive model* provides a more detailed description of the system by breaking it down into a set of related independent and dependent variables, i.e. an underlying theoretical model. Kuechler and Vaishnavi (2012) defined this design component as a mid-range theory in which cause-and-effect relationships in *kernel knowledge* are translated into a design-relevant context. This is similar to a traditional theoretical model, with the exception that the level of detail may be less, causing some proponents of design-oriented approaches to liken it to Weick's (1995) description of 'theorizing' (Baskerville and Pries-Heje 2010; March and Smith 1995; Orlikowski and Iacono 2001). Winter (2008, p.471) argued that for a design endeavor to be faithfully termed 'research', this "implies that problem solutions should be generic to some extent, i.e., applicable to a set of problem situations". It is this theoretical abstraction of the problem space that embodies the generality of the problem, thus differentiates a piece of design research from industrial design by ensuring the reusability of findings (vom Brocke and Buddendick 2006). An example of this is evident in Parsons and Wand (2008), who developed design principles for the classification of system components to facilitate better modeling of information systems. In order to describe system modeling (i.e. the problem system) and inform design, Parsons and Wand defined the classification problem according to three design questions, i.e. what constitutes a class, how do we identify classes, and when should subclasses be defined? These three independent variables form the target for the actual design theory (i.e. the proposed classification principles), which seeks to address them to impact upon system modeling (i.e. the implied dependent variable). Just as the demonstration of utility validates the high-level assumptions underlying the *utility requirements*, the more detailed description of the problem system provided by the *explanatory/predictive model* is also validated.

The constructs and relationships in the *explanatory/predictive model* are referred to separately by Hevner et al. (2004) as 'constructs' and a 'model' and by Carlsson et al. (2011) as 'problem situation' and 'mechanism'. Hevner et al. describes 'constructs' as the components that "provide the language in which problems and solutions are defined and communicated", and 'models' as components that "use constructs to represent a real world situation the design problem and its solution space" (2004, p78). This is in contrast to the remaining sources, each of which combine this cause/effect breakdown of the problem domain under the single headings of 'meta-design' (Walls et al. 1992), 'principles of form and function' (Gregor and Jones 2007), 'objectives of a solution' (Peffers et al. 2008), 'suggestion (Kuechler and Vaishnavi 2008), 'general components' (Baskerville and Pries-Heje 2010), 'system architecture' (Nunamaker et al. 1990) and 'problem identification' (Cole et al. 2005). Kuechler and Vaishnavi (2012) emphasize the role of this design component in translating technology-free relationships into a technological design-oriented domain. They argue that constructs are transformed into their design-relevant form on the basis of some cause/effect relationship with target phenomena. Based upon this, it is not meaningful to discuss constructs prior to establishing this cause/effect relationship, as this reflects the basis for their inclusion. Thus the explanatory/predictive model is referred to as a single component in this study, as to do otherwise would impose a logically impossible sequence upon the process model.

As discussed already within the section entitled 'Design-oriented research in IS', in this study both the *procedural transparency of the design process* and the broader *theoretical rigor of design process* are presented as independent variables in determining the *perceived rigor of design process*, which subsequently contributes to the *impact of DSR*. These constructs, as well as the cause/effect relationships connecting them, are depicted in Figure 2.

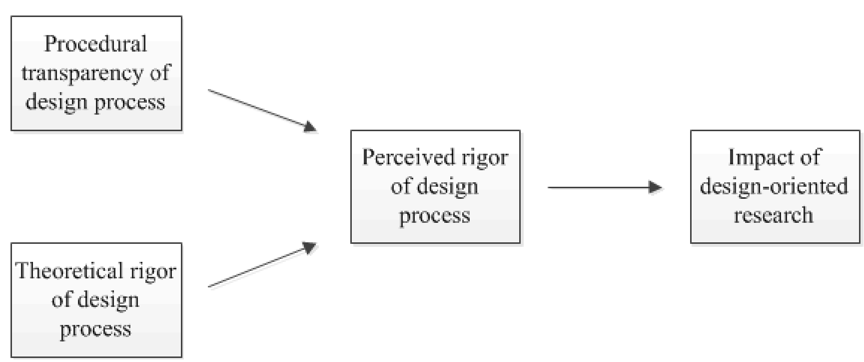

Figure 2. Explanatory/predictive model for the design process theory

The development of the design theory

The *design theory* represents the actual design prescriptions intended to impact upon the problem system, or "goal-directed plans for manipulating constructs" (Vaishnavi and Kuechler 2007, p.13). This relationship with the *explanatory/predictive model* was also reinforced by March and Smith (1995, p.257), who defined methods as "a set of steps (an algorithm or guideline) used to perform a task. Methods are based on a set of underlying constructs (language) and a representation (model) of the solution space". March & Smith (1995) went on to clarify that this relationship exists even when not made explicit by researchers, remarking that "although they may not be explicitly articulated, representations of tasks and results are intrinsic to methods" (p.257). Hevner et al. (2004, p.79) states that design theories "can range from formal, mathematical algorithms that explicitly define the search process to informal, textual descriptions of best practice approaches, or some combination". Goldkuhl (2004, p.63) provided a more reductionist perspective, stating that prescriptive statements are essentially reducible to the format of "Perform act A in order to obtain goal B" (a structure that the authors of this study compare to that of Van Aken's (2005) 'technological rules'). Regardless of their exact format, this is the applicable design component that affords the actual utility of design-oriented research. Moreover, this practically implementable design component endows design-oriented research a valuable additional level of industrial relevance (c.f. Benbasat and Zmud 1999). An example of such a *design theory* can be seen in Markus et al. (2002) who developed a design theory comprising six design principles, the application of which supported emergent knowledge processes. Each of these design principles provided concrete prescriptions for industrial practitioners with tangible and validated 'how to' findings, which could be adopted in practice with a minimum of interpretation. Accordingly, the quality of these design principles was determined according to their ability to impact upon emergent knowledge processes.

This design component is described in similar terms by each source. This reference comes under the headings 'method' (Hevner et al. 2004), 'design method' (Walls et al. 1992), 'principles of implementation' (Gregor and Jones 2007), 'design and development' (Peffers et al. 2008), 'design theory' (Kuechler and Vaishnavi 2008), 'design practice theory' (Baskerville and Pries-Heje 2010), 'system design' (Nunamaker et al. 1990), 'intervention' (Cole et al. 2005), and 'design propositions' by Carlsson et al. (2011). The *design theory* developed in this study is the prescriptive Process Model for Procedural Transparency (PMPT). This process model centers upon nine design components identified by an analysis of the eight primary literature sources already identified during the *development of kernel knowledge*. The theorizing process for this process model was similar to that of Peffers et al. (2008), in that the eight primary literature sources were analyzed via an inductive and abductive process. This analytic process required that each design component described by each of the sources was first noted independently, and then added to an exhaustive list maintained of all design components identified across the eight sources. Design components were then compared to those observed in other sources and the components that were found to be analogous at a high-level were combined into one generalized component. The final set of design components is represented by the leftmost column in Table 1. It is noteworthy that unlike Peffers et al., the criteria for inclusion of a design component in this study demanded that it represents an important part of the theorizing process, rather than just an important part of the finished design product.

Two exceptions to this theorizing process were implemented. Firstly, Hevner et al. (2004) and Carlsson et al. (2011) each decomposed the *explanatory/predictive model* down into what can be termed 'explanatory/predictive constructs' and 'relationships between explanatory/predictive constructs'. Secondly, Walls et al. (1992) decomposed *Kernel knowledge* into 'design product kernel theory' and 'design process kernel theory'. For reasons presented in the discussion of the *development of kernel knowledge* and the *development of the explanatory/predictive model* respectively, the authors of this paper felt that maintaining these subdivisions would result in design components that could not meaningfully be discussed in isolation from one another. Hence a combined view of these components was maintained.

The development of the instantiation

Following the development of the *design theory*, which describes how to manipulate the *explanatory/predictive model*, an *instantiation* may be developed. This *instantiation* implements the *design theory* in a system and represents the transition of the design prescriptions from the abstract into a real setting. This transition affords two central responsibilities to an *instantiation*. The first of these is to demonstrate that the *design theory* can in fact be instantiated, as "In much of the computer science literature it is realized that constructs, models, and methods that work 'on paper' will not necessarily work in real world contexts. Consequently, instantiations provide the real proof" (March and Smith 1995, p.260). The second responsibility of the *instantiation* is to demonstrate *how* the prescriptions can be implemented. This can be seen in the example of Chaturvedi et al. (2011), who developed a *design theory* in the form of two complimentary sets of design principles for the design of virtual worlds. Chaturvedi et al. demonstrate the *instantiation* of these principles in the form of an exemplar large-scale mirror virtual world, in which they had been involved with the development. The provision of such an example not only proves the instantiability of the design prescriptions, it also serves to make their intent more lucid for designers and reduce the potential for misapplication (Tetzlajf and Schwartz 1991). This point was also made by Gregor and Jones (2007, p.329) who remarked that, "*To take an example, the placement of items on a computer screen can be described using screen coordinates. This process is tedious, and the results are not very understandable. A copy of a screen display is more immediately comprehended and would serve better if one were illustrating some guidelines for a screen design.*"

Table 1. Components/outputs of a design theory identified in existing research and the terms used to describe them.

	Walls et al. (1992, 2004)	Gregor and Jones (2007)	Hevner et al. (2004)	Peffers et al. (2008)	Baskerville and Pries-Heje (2010)	Nunamaker et al. (1990)	Kuechler and Vaishnavi (2008)	Carlsson et al. (2011)
Utility requirements	Meta-requirements	Purpose and scope	Utility	Problem identification /motivation	General requirements	Conceptual framework	Awareness of problem	Outcome
Kernel knowledge	Design product kernel theory Design process kernel theory	Justificatory knowledge	Applicable knowledge		Capability		Kernel theory	Extant knowledge, data, theories
Explanatory/ predictive model	Meta-design	Principles of form and function	Constructs Model	Objectives of a solution	General components	System architecture	Suggestion	Problem situation Mechanism
Design theory	Design method	Principles of implementat-ion	Method	Design and development	Design practice theory	System design	Design theory	Design propositions
Design iterations		Artifact mutability	Design iterations	Process iteration		Experiment	Circumscript-ion	Propose/ refine design theory
Instantiation	Two-phase testing	Expository instantiation	Instantiation	Demonstrat-ion		Build the system	Development	Alpha testing
Design process evaluation	Testable design process hypotheses	Testable propositions						
Utilitarian evaluation	Testable design product hypotheses		Evaluation	Evaluation		Observe and evaluate	Evaluation	Test design theory and knowledge
Additions to knowledge			Additions to knowledge	Communicat-ion			Conclusion	Propose/ refine knowledge

This component is referred to as an 'instantiation' (Hevner et al. 2004), 'an expository instantiation' (Gregor and Jones 2007), 'demonstration' (Peffers et al. 2008), 'development' (Kuechler and Vaishnavi 2008), or 'building the system' (Nunamaker et al. 1990). Other descriptions are less explicit. Cole et al. (2005) capture both the instantiation and method under the title of an 'intervention', whereas Walls et al. (1992) and Carlsson et al. (2011) describe the component implicitly in terms of theory-testing, as part of 'two-phase testing' and 'alpha testing' respectively. Baskerville and Pries-Heje (2010) do not acknowledge this design component at all, reflecting their self-described 'explanatory', rather than 'practice' focus.

The process model developed in this study is instantiated retroactively to Markus et al.'s (2002) development of a design theory to support emergent knowledge processes. This study is selected as it represents a well-cited and informative piece of design-oriented research published in an international peer-reviewed journal. Furthermore, the design process adopted by Markus et al. is iterative and complex, and so represents a non-trivial challenge for PMPT. It is noteworthy that Markus et al. present the study as 'design theory', rather than DSR. However, consistent with the combined perspective of DSR and design theory research adopted throughout this paper, Markus et al. is nonetheless an appropriate platform for instantiating the process model. Space limitations in this paper demand that an abridged description of Markus et al.'s design process is presented, hence a high-level account is presented in Table 2. This transformation represents a subjective interpretation of the design process used by Markus et al. and it is acknowledged that different interpretations of some aspects of the description are possible, particularly with regard to the development of the explanatory/predictive model. This is not presented as a limitation but rather as part of the demonstration that opportunities exist for greater transparency in some aspects of Markus et al.'s account of the design process.

Table 2. Instantiation of PMPT retroactively to Markus et al. (2002).

Design process	*Summary of description*
Development of utility requirements	Theoretically grounded design principles that guide the development of decision support systems, expert systems and executive support systems
Development of kernel knowledge	Theoretical and practical IS literature surrounding decision-support systems, expert systems, and executive information systems.
Development of explanatory/ predictive model	Causes: (1) identification of target group of users, (2) achieving consensus over requirements, (3) incorporation of experts' knowledge on how task is best performed and what information is relevant. Effects: Effective system design
Development of design theory	(1) System must support the consensus needs of a known user community, determined through use of a method like any time RAD, (2) System must represent experts' knowledge of organization design as if-then rules with prescriptions for action, (3) Knowledge-base should provide a single, cross-functional view the organization, (4) System must incorporate the prescribed process and restrict ad hoc organization designers to the accepted process.
Development of instantiation	Unspecified prototypes testing various assumptions deployed as part of the Technology, People and Organization integration (TOP) modeler project involving Hewlett-Packard, General Motors, Digital Equipment Corporation, and Texas Instruments.
Development of utilitarian evaluation	Observations of user responses failed to support principles; although it is unspecified as to what exact criteria principles were evaluated against. However, Markus et al. describe that these observations revealed that no single identifiable user group existed for many knowledge processes. Markus et al. also report that lay organization designers circumvent prescribed processes, do not use rules, do not follow prescriptions for action and do not seek input from other groups.
Development of design iterations	Reconsideration of system design concept to one that embraces less rigid aspects of organizational structure.

Development of kernel knowledge (revised)	Additional focus on literature describing non-sequential deliberation processes, unknown user types in strategic planning, and user information search among poorly organized sources. Definition of the concept of Emergent Knowledge Processes (EKPs) to characterize these processes.
Development of explanatory/ predictive model (revised)	Causes: (1) System does not target specific user roles, depend on training, or assume motivation to use the tool, (2) System accommodates complex, distributed, and evolving knowledge-bases, (3) System supports an unstructured, dynamically changing process of deliberations and tradeoffs Effects (first-order): Effective EKP support system, (second-order): Effective system design
Development of design theory (revised)	(1) Design for customer engagement by seeking out naïve users, (2) Design for knowledge translation through radical iteration with functional prototypes, (3) Design for offline action, (4) Integrate expert knowledge with local knowledge sharing, (5) Design for implicit guidance through a dialectical development process, (6) Componentize everything, including the knowledge-base.
Development of instantiation (revised)	TOP modeler was adapted to include features such as: the 'Ferris wheel' to draw in naïve users and facilitate dialectical development; a 'tradeoff matrix' as a radical iteration of design to translate knowledge; a 'summary gap panel' and 'design model' to encourage offline action; dialogue-supporting visualizations and forms to afford knowledge sharing and; a modularly defined breakdown of the TOP system to ensure components were compartmentalized.
Development of utilitarian evaluation (revised)	Observations of user responses supported new system, however again it is unspecified as to what these observations were as well as against which criteria principles were evaluated.
Development of additions to knowledge	The definition and illumination of the concept of EKPs, as well as a design theory theorizing how system features can be integrated to support them.

The development of the utilitarian evaluation

Once an iteration of the *design theory* has been completed and *instantiated*, the *utilitarian evaluation* can begin. This utilitarian evaluation seeks to implement the *design theory* and observe the impact on the dependent variable in the system. Thus, in essence this is where the design process introduces an empirical element to parallel the empirical validation in traditional research approaches (March and Smith 1995). A number of methods have been suggested for this evaluation, including case studies, field studies, and various analytical, experimental, testing and descriptive methods (Hevner et al. 2004). Peffers et al. (2008, p.56) summed this up by stating *"This activity involves comparing the objectives of a solution to actual observed results from use of the artifact in the demonstration. It requires knowledge of relevant metrics and analysis techniques… Conceptually, such evaluation could include any appropriate empirical evidence or logical proof."*

Given that the *development of the instantiation* has already presented a re-contextualization of the *design theory,* the utilitarian evaluation may choose to build upon the *instantiation*. Such a cumulative step is natural provided the *instantiation*'s representation of the *design theory* is both accurate and well defined. March and Smith (1995, p.261) commented that "a difficulty with evaluating instantiations is separating the instantiation from the constructs, models, and methods embodied in it". For example Lee et al. (2008) developed a template for business process design, which laid out a grammar-based method for identifying and managing potential business processes. As part of the design process, Lee et al. instantiated a software prototype which embodied the design prescriptions, generating business process on the basis of an inputted grammar. The performance of this software prototype in the design of a sales process was subsequently used as a means of evaluation. However Lee et al. argue that further field studies in various real organizations are needed in order ensure the *utilitarian evaluation* is adequately broad to validate the template proposed. In another example, Adomavicius et al. (2008) developed an IT ecosystem model as a *design theory* to predict the return on organizational technology investment. Adomavicius et al.

instantiated the *design theory* within two separate contexts, digital music technologies and Wi-Fi technologies and the utility of the *design theory* was subsequently evaluated qualitatively across both environments, allowing for a triangulation of instances.

The *utilitarian evaluation* of the IT Artifact is referenced as a design component by all but one of the sources, namely Baskerville and Pries-Heje (2010). This component is referenced as 'testable design product hypotheses' (Walls et al. 1992), 'testable propositions' (Gregor and Jones 2007), 'observation and evaluation' (Nunamaker 1990), 'testing design theory and knowledge' (Carlsson et al. (2011), and most commonly 'evaluation' (Hevner et al. 2004, Peffers et al. 2008, Kuechler and Vaishnavi 2008, Cole et al. 2005). The frequency of this design component is not surprising, given the role of this component in both confirming the utility of the design theory and empirically validating the theoretical contribution.

The utilitarian evaluation of PMPT builds upon the instantiation of Markus et al. (2002) in the previous section. Because the process model is applied retroactively, its capacity to facilitate procedural transparency, i.e. its utility, is evaluated by considering the opportunities it identifies to improve the *procedural transparency* of the sample article. In addition to the comments in the previous section regarding the subjective interpretations required to explicate the explanatory/predictive model, opportunities for increased procedural transparency in the case of Markus et al. emerge in two parts of PMPT: (1) The instantiation of prototypes to test various assumptions in the first iteration of design. Markus et al. (2002, p.188) stated that "*the development team repeatedly intervened into the organizational design activities of the involved companies, deploying prototypes that tested various assumptions about how organizational design work is done, observing how users responded, and iterating. Finally, we articulated our learning in the form of a new IS design theory*". It is implied that these assumptions manifested the principles comprising the initial design theory, however Markus et al. do not specify what these evaluator prototypes were or how they were presented to users. As part of the contribution from their study relates to the discovery that these assumptions are flawed, a more explicit account of how their disconfirmation would add to the *perceived rigor* of the study. (2) The utilitarian evaluation performed on the final design theory. This evaluation is carried out qualitatively by Markus et al. but it is not stated exactly how they formed their judgments of utility. For example, the authors describe support for Principle #3 by stating "Using functional prototypes allowed us to intervene directly in the work process and observe which aspects of the system worked and which did not" (p.196). However it is not clear whether the system 'working' implies that that users are willfully adopting the system into their processes, that some increased procedural efficiency is observed from users adopting the system, that management has expressed satisfaction with the system, or there is some other evidence that it is tangibly supporting EKPs. The accounts given for the emergence of each principle include a variety of positive indicators for several of these possibilities. However the embedded nature of the discussion of the *utilitarian evaluation* makes it less obvious what the ultimate unifying measurement of utility is for the study, hence somewhat erodes the *perceived rigor* of the evaluation and consequently the overall study.

The retroactive application of PMPT to Markus et al. (2002) identifies two opportunities for increased *procedural transparency* and consequently increased *perceived rigor*. In light of the earlier discussion regarding the importance of *perceived rigor* on the impact of design-oriented research, it must be concluded that had Markus et al. applied PMPT to describe their research, it is likely that the impact of their study would have been even greater. Thus, the utilitarian evaluation of PMPT confirms the utility of the process model, based on the opportunities for increased *procedural transparency* that have been observed.

The development of design iterations

A further component of design reflects the incremental and iterative nature of developing a 'method', which represents a fundamental feature of design (c.f. Simon 1996). Gregor and Jones (2007, p.326) strongly advocated the iterative aspect of design, stating that "*the lack of theories about IT artifacts, the ways in which they emerge and evolve over time, and how they become interdependent with socio-economic contexts and practices, are key unresolved issues for our field and ones that will become even more problematic in these dynamic and innovative times*". An example of the effectiveness of *design theory iterations* is observed in Kolfschoten and de Vreede (2009). Kolfschoten and de Vreede developed a process model to facilitate collaboration between system designers and users, without the arbitration of third parties. This *design theory* was evaluated over four phases of data gathering and analysis, punctuated by periods of reflection based on results and feedback, and continuous redesign. Although the *design theory* was initially developed on a base of existing literature, it became gradually more empirically driven over the course of development.

This description of iterative design-oriented research raises an interesting philosophical issue surrounding design-oriented research, as it emphasizes the role of *induction*, i.e. when theoretical understanding results from empirical observations (Popper 2002). DSR in particular is commonly performed within a modern positivist epistemology (Cross 2001; Love 1998; McKay and Marshall 2005; Vaishnavi and Kuechler 2007), yet this inductive component represents a significant departure from the hypothetico-deductive methods traditional to positivistic inquiry (Straub et al. 2005). Traditional opposition to inductive scientific research traces its roots back to Karl Popper's 1969 book *Conjectures and Refutations* (reprinted 2002). Popper argued for a process of 'repetition by deduction', in order to test the accuracy of the predictions offered by theoretical perspectives. According to this perspective, while induction and introspection are important, it is in the role of "pre-science; that is, as a stage leading to the creation of scientific theory" (Straub et al. 2005, p.227). However, Coombs (1960) highlighted a fundamental flaw with this understanding when it was demonstrated that all measurement is premised upon some theoretical assumptions, i.e. it is 'theory-laden'. Consequently, if induction occurs as a matter of pre-science and this induction must always be performed on the basis of some theory, which is to say it must follow some other deduction; then induction and deduction must lie at alternating points of a reciprocal scientific cycle. Thus, the authors of this study propose that design-oriented research differs from traditional hypothetico-deductive approaches as a matter of emphasis rather than inclusion, as a significant inductive component to any scientific inquiry is logically inescapable. This observation is significant, as it maintains the epistemological freedom of design-oriented research.

This decreasing allegiance to purely hypothetico-deductive methods can also be witnessed in the growth of positivistic case-study research (Lee 1991; Yin 2008) and has been related to ontological changes in positivism over the past few decades, specifically the growth of 'critical realism' (Mingers 2004; Straub et al. 2005). The critical realist perspective was summed up succinctly by Carlsson et al. (2011, p.113), who remarked, "Critical realism takes an objective ontology and a subjective epistemology". Indeed it has been suggested that critical realism presents the natural philosophical backdrop for design-oriented research (Bunge 1985; Carlsson 2006; Vaishnavi and Kuechler 2005). Whether or not this is the case is a matter of contention (c.f. Cross 2001; Gregg 2001), however modern positivism continues to house much of the design-oriented research in IS (McKay and Marshall 2005).

This *design iteration* component is reasonably pervasive in most of the sources. Hevner et al. (2004), Gregor and Jones (2007), Vaishnavi and Kuechler (2008), Peffers et al. (2008) and Nunamaker et al. (1990) refer to this component as 'design iterations', 'Artifact mutability', 'circumscription', 'process iterations', and 'experimenting', respectively. Cole et al. (2005) also allude to the iterative nature of design when they refer to the cyclical process of 'interventions', as do Carlsson et al. (2011) by positioning the 'propose/refine design theory' within the loop-based section of their framework. Neither Walls et al. (1992) or Baskerville and Pries-Heje (2010) describe this design component in any detail, however Walls et al. do include the iterative design component in their design exemplars, suggesting that it is implied as part of the design process.

The instantiation of PMPT illustrated one possible additional design stage emerging from the description of the design theory presented in Markus et al. (2002, p. 206). Under the heading of 'kernel theory' Markus et al. list three foremost characteristics of emergent knowledge processes. These three characteristics resemble the concept of 'social mechanisms', i.e. the formative and causal explanatory mechanisms that support a range of theoretical relationships (c.f. Hedström and Swedberg 1996). This concept resonates with the account of the *explanatory/predictive model* presented by Kuechler and Vaishnavi (2012), in which relevant causal relationships are identified from *kernel knowledge* are translated into a design context. As a result of this, the question arises whether a further design stage should be included, i.e. that of *the development of kernel mechanisms*. The decision is made here not to include this possible new design component. This is because the identification of these mechanisms is already the core function of *the development of kernel knowledge*, even if this function was not explicit prior to the observations from Markus et al. (2002). As such, this observation represents an enrichment and clarification of this existing design component, rather than a need to decompose it into two separate components. Were such a decomposition implemented the *development of kernel knowledge* would refer only to the nominal identification of relevant literature, an account of which in isolation would add questionable value.

The development of additions to knowledge

As discussed already, design prescriptions may contribute to theory by validating or invalidating the theoretical basis for design, such as that provided by the *explanatory/predictive model* and *kernel knowledge*. Furthermore, previously unrecognized relationships between constructs may emerge from the manner in which a *design theory* functions, in particular as successive design theory iterations explore unanticipated findings. An example of this can be seen in McLaren et al. (2011) who developed a Multilevel Strategic Fit model prescribing seven steps for businesses to measure the fit of their IS capabilities with their strategic needs at various organizational levels. The development of this multilevel model required the conceptualization of organizations in a way that was both measureable and representative of the actual organizational structure. As such, when the proposed model demonstrated utility as a means of determining multilevel fit, McLaren et al.'s conceptualization of organizations was also validated. As a result, McLaren et al. report a clearer theoretical understanding of certain ill-defined organizational components, such as a number of those involved with Supply Chain Management. Vaishnavi and Kuechler (2008, p.496) described the increased exploratory capability of design-oriented research over traditional methods as follows:

"*In natural science research the experimental procedure and apparatus are (ideally) constructed in such a way as to minimize confounds that might interfere with clear interpretation of the results; theory is either supported or disconfirmed. In design science research both the artifact and the experimental setting are intentionally complex (and thus confounded) in order to develop methods and artifacts that are useful in practice… This fundamental difference encourages the iteration between design and evaluation that would be considered improper 'fishing for data' in a natural science experiment.*"

Accordingly, this design component features in each of the DSR sources as 'additions to knowledge' (Hevner et al. 2004), 'communication' (Peffers et al 2008), 'conclusions' (Kuechler and Vaishnavi 2008) and 'propose/refine knowledge' (Carlsson et al. 2011). It is also referred to by Cole et al. (2005) as reflection and learning'. However, it is not discussed in Walls et al. (1992), Gregor and Jones (2007), Baskerville and Pries-Heje (2010) or Nunamaker et al. (1990). This resonates with observations by Winter (2008) that the 'reflection' of findings back to existing theoretical literature is what differentiates DSR from other design-oriented research. Similarly Walls et al. also argued that the development of design theories represents one aspect of the greater DSR paradigm, remarking "we do not see our view of design theory to be in conflict with the design science perspective but rather complimentary to and an integral part of that perspective" (2004, p.48).

The *additions to knowledge* from this study arise from three main findings. Firstly, an argument for increased procedural transparency is put forward as a means of increasing the impact of DSR. Secondly, an IS-specific overview and comparison of design-oriented literature is presented according to the design components described by existing literature sources. These two findings are significant in their own right, as they represent the definition of the problem space and resonate with Baskerville and Pries-Heje's (2010) definition of an *explanatory design theory*. Thirdly, a process model is presented as a prescriptive *design theory* to facilitate *procedural transparency* in future DSR studies, so increasing their perceived rigor and thus their impact. This finding is perhaps the most significant, as it presents IS researchers with a tool that is both tangible and valuable for application in future research.

The development of design process evaluation

In keeping with the emphasis placed upon the design process by Walls et al. (1992), these authors also argued that design-oriented research should include 'testable design process hypotheses' as well as 'testable design product hypotheses'. Walls et al. (1992, p.43) describe these design components as "used to verify whether the design theory results in an artifact which is consistent with the meta-design". The *design process evaluation* component of the process model is different to other design components, as it features at more than one stage of development. This is because the design process can, and should be, evaluated on a continuous basis as the early identification of design faults impacts significantly upon the both the cost and likelihood of their remediation (Holzinger, 2005). Perhaps unsurprisingly given the controversy surrounding the rigor of the design process, Walls et al. (1992) is the only source to identify *design process evaluation* as a separate design component. Similar recommendations are subsumed under the headings of 'testable propositions' by Gregor and Jones (2007) and 'evaluation' by Cole et al. (2005). Kuechler and Vaishnavi (2008) acknowledge the recommendations to identify this component made by Walls et al. (1992, 2004), however declined to comment further than to remark that their own development process was not novel in any way. Interestingly, the *design process evaluation* design component is not discussed directly in any of the other sources. Moreover, this represents the only neglected design component by all four frameworks corresponding to sources that explicitly refer to themselves as DSR, i.e. Hevner et al. (2004), Peffers et al. (2008), Kuechler and Vaishnavi (2008), and Carlsson et al. (2011).

The goal of PMPT is the *procedural transparency of the design process*, thus the *design process evaluation* in this instance manifests the testable hypothesis that an explicit account of each design component is presented. This has also been demonstrated at a recursive level during the development of PMPT, as part of which each design component of PMPT has been described.

Conclusions and future work

This study has presented a process model to increase the *procedural transparency of the design process* in DSR studies. This process model makes the various stages of development evident in DSR studies explicit, and brings to light key issues during each stage of the design process. Furthermore, this process model differs from previous models, such as that of Peffers et al. (2008) whose central goal was to provide a standardized template for DSR. Instead, the process model developed in this study seeks to increase the *perceived rigor* of a study by increasing the visibility of aspects of the design process that may otherwise go undocumented. Thus, the applications for these models are significantly different. For example, an *ex post* evaluation of existing research with a model such as Peffers et al.'s may be used to break the research down into familiar sections, ensuring studies are compared according to the same components. An *ex post* evaluation of existing research with PMPT breaks down the research into familiar sections and identifies what aspects of the design process, if any, the authors have failed to document explicitly. Similarly, a DSR endeavor carried out within the framework of the DSRM will be communicable within a standard grammar and lexicon, reducing the potential for misappropriation. A DSR endeavor carried out within the framework of the process model developed in this study will also be communicable within a standard grammar and lexicon, however it will also be *procedurally transparent* in its representation of the design process. This study has also implemented PMPT, both at a recursive level, as well as retroactively on a piece of high quality extant research. These implementations further validated the consistency, practicability and usefulness of the process model by demonstrating its capacity to afford *procedural transparency*.

PMPT has two key limitations that present avenues for future work. Firstly, the adoption of PMPT in future research embedded within specific contexts is necessary. Such a step is required in order to investigate the *procedural transparency* afforded by PMPT under different environmental conditions. Such varying environmental conditions would present one of three outcomes, each of which is valuable to researchers. In the first outcome, PMPT may accurately describe a study's proposed design process, adding *procedural transparency* without encountering any significant issues. In this case, PMPT is further validated and the study in question benefits from added *perceived rigor*. In the second outcome, PMPT may conflict with a study's proposed design process in a way that highlights issues with that study's design process. This would also be a valuable finding, as the prescriptive power of PMPT would be further validated and an inappropriate design process avoided by the study in question. In the third outcome, PMPT may conflict with a study's proposed design process in a way that highlights issues with PMPT. This would present an opportunity for refinement of PMPT, as well as some re-evaluation of the understanding that underlies it. It is also possible that the second and third outcome may arise in tandem, in which case opportunities for improvement would arise for both PMPT and a study's proposed process model.

The second limitation surrounds the limited scope of PMPT. The process model developed in this study targeted *procedural transparency of the design process*, as this represents a concern for rigor that transcends specific philosophical or methodological circumstances. Nevertheless, the broader issue of the *theoretical rigor* of the design process remains an important contributing factor to the perceived rigor of DSR. To this end, the analysis of design-oriented literature revealed a number of design component-specific issues relating to the more functional aspects of *theoretical rigor* that are equally independent of philosophy or methodology, e.g. the intellectual trajectory of theories in the *development of kernel knowledge*, or the measurement of appropriate variables in the *development of the utilitarian evaluation*. These present an opportunity for the compilation of design component-specific issues to inform separate *design process evaluation* testable hypotheses at each stage of development. For example, it was acknowledged in the discussion of the *development of utility requirements* that a DSR study must focus upon 'wicked problems' in order to ensure a significant theoretical contribution. Hevner et al. (2004) characterized such problems according to five criteria (1) Unstable requirements and constraints based upon ill-defined environmental contexts (2) Complex interactions among subcomponents of the problem and its solution (3) Inherent flexibility to change design processes as well as design artifacts (4) A critical dependence upon human cognitive abilities to produce effective solutions (5) A critical dependence upon human social abilities to produce effective solutions. Such criteria, in addition to the existing requirements surrounding transparency, could therefore be applied to the *design process evaluation* for the *development of utility requirements*.

We thus call for two types of future research. Firstly, we call for DSR studies employing PMPT. These studies are likely to benefit IS by providing the community with DSR of a higher *perceived rigor*, hence of more impact, as well as offering the capacity to refine PMPT in the event that the proposed description of the design process is found to be inadequate. Secondly, we call for future research developing a DSR process model that provides concrete evaluation prescriptions for the *design process evaluation* for each stage of design. Such a process model would provide IS scholars with an increasingly detailed, repeatable and objective template for DSR research and presents the next natural step in the 'theorizing' of DSR methodological theories.

References

Adipat, B., Zhang, D., Zhou, L. 2011. "The effects of tree-view based presentation adaptation on mobile web browsing." MIS Quarterly (35:1), pp. 99-122.

Adomavicius, G., Bockstedt, J. C., Gupta, A., Kauffman, R.J. 2008. "Making sense of technology trends in the information technology landscape: a design science approach." MIS Quarterly (32:4), pp. 779-809.

Archer, L. B. 1984. *Systematic method for designers. Developments in design methodology*. N. Cross. London, John Wiley, pp. 57-82.

Baskerville, R., Pries-Heje, J. 2010. "Explanatory Design Theory." Business & Information Systems Engineering (2:5), pp. 271-282.

Benbasat, I., Goldstein, D. K., Mead, M. 1987. "The case research strategy in studies of information systems." MIS Quarterly (11:3), pp. 369-386.

Benbasat, I., Zmud, R.W. 1999. "Empirical research in information systems: the practice of relevance." MIS Quarterly (23:1), pp. 3-16.

Benbasat, I., Zmud, R.W. 2003. "The identity crisis within the IS discipline: Defining and communicating the discipline's core properties." MIS Quarterly (27:2), pp. 183-194.

Boudreau, M.C., Gefen, D., Straub, D.W. 2001. "Validation in Information Systems Research: A State-of-the-Art Assessment." MIS Quarterly (25:1), pp. 1-16.

Bunge, M.A. 1985. *Philosophical inputs and outputs of technology. History and Philosophy of Technology*. G. Bugliarello and D. B. Doner. Urbana, Illinois, University of Illinois press, pp. 263-281.

Carlsson, S.A. 2006. "Towards an Information Systems Design Research Framework: A Critical Realist Perspective," in *Proceedings of the First International Conference on Design Science in Information Systems and Technology* (DESRIST 2006), Claremont, 2006, pp. 192-212. Carlsson, S. A.,

Carlsson, S.A., Henningsson, S., Hrastinski, S., Keller, C. 2011. "Socio-technical IS design science research: developing design theory for IS integration management." Information Systems and e-Business Management (9:1), pp. 1-23.

Chaturvedi, A.R., Dolk, D.R., Drenevich, P.L. 2011. "Design principles for virtual worlds." MIS Quarterly (35:3), pp. 673-684.

Cole, R., Purao, S., Rossi, M. and Sein, M.K., 2005. "Being Proactive: Where Action Research Meets Design Research," *Proceedings of the Twenty-Sixth International Conference on Information Systems*, Las Vegas, 2005, pp. 325-336.

Coombs, C. H. 1960. "A Theory of Data." Psychological Review (67:3), pp. 143-159.

Cross, N. 2001. "Designerly ways of knowing: design discipline versus design science." Design Issues (17:3), pp. 49-55.

Dubé, L., Paré, G. 2003. "Rigor in Information Systems Positivist Case Research: Current Practices, Trends, and Recommendations." MIS Quarterly (27:4), pp. 597-636.

Eekels, J., Roozenburg, N.F.M. 1991. "A methodological comparison of the structures of scientific research and engineering design: their similarities and differences." Design Studies (12:4), pp. 197-203.

Fallman, D. 2003. *Design-oriented human-computer interaction*, ACM New York, NY, USA.

Frintrop, S., Rome, E., Christensen, H.I. 2010. "Computational visual attention systems and their cognitive foundations: A survey." ACM Transactions on Applied Perception (7:1), pp. 1-39.

Goldkuhl, G. 2004. "Design theories in information systems-a need for multi-grounding." Journal of Information Technology (6:2), pp. 59-72.

Gregg, D.G., Kulkarni, U.R., Vinzé, A.S. 2001. "Understanding the philosophical underpinnings of software engineering research in information systems." Information Systems Frontiers (3:2), pp. 169-183.

Gregor, S., Jones, D. 2007. "The anatomy of a design theory." Journal for the Association of Information Systems (8:5), pp. 312-335.

Hanseth, O., Lyytinen, K. 2010. "Design theory for dynamic complexity in information infrastructures: the case of building internet." Journal of Information Technology (25:1), pp. 1-19.

Harrison, W. L., Farn, C.K. 1990. "A comparison of information management issues in the United States of America and the Republic of China." Information & Management (18:4), pp. 177-188.

Hedström, P., Swedberg, R. 1996. "Social mechanisms." Acta Sociologica (39:3), pp. 281-308.

Hevner, A. R. 2007. "The three cycle view of design science research." Scandinavian Journal of Information Systems (19:2), pp. 87.

Hevner, A. R., March, S.T., Park, J., Ram, S. 2004. "Design Science Research in Information Systems." MIS Quarterly (28:1), pp. 75-105.

Holzinger, A. 2005. "Usability engineering methods for software developers." Communications of the ACM (48:1), pp. 71-74.

Iivari, J. 2007. "A paradigmatic analysis of information systems as a design science." Scandinavian Journal of Information Systems (19:2), pp. 39-64.

Klein, H. K., Myers, M.D. 1999. "A set of principles for conducting and evaluating interpretive field studies in information systems." MIS Quarterly (23:1), pp. 67-93.

Kolfschoten, G.L., de Vreede, G.J. 2009. "A Design Approach for Collaboration Processes: A Multimethod Design Science Study in Collaboration Engineering." Journal of Management Information Systems (26:1), pp. 225-256.

Kuechler, B., Vaishnavi, V. 2008. "On theory development in design science research: anatomy of a research project." European Journal of Information Systems (17:5), pp. 489-504.

Kuechler, B., Vaishnavi, V. 2012. "A Framework for Theory Development in Design Science Research: Multiple Perspectives," Journal of the Association for Information Systems (13:6), pp. 395-423.

Lee, A.S. 1991. "Integrating positivist and interpretive approaches to organizational research." Organization Science (2:4), pp. 342-365.

Lee, J., Wyner, G.M., Pentland, P.T. 2008. "Process grammar as a tool for business process design." MIS Quarterly (32:4), pp. 757-778.

Love, T. 1998. *Social, environmental and ethical factors in engineering design theory: a post positivist approach*, Praxis Education, Perth, WA.

March, S. T., Smith, G.F. 1995. "Design and natural science research on information technology." Decision Support Systems (15:4), pp. 251-266.

Markus, M. L., Majchrzak, A., Gasser, L. 2002. "A design theory for systems that support emergent knowledge processes." MIS Quarterly (26:3), pp. 179-212.

McKay, P., Marshall, J. 2005. "A review of design science in information systems," in *Proceedings of the 16th Australasian Conference on Information Systems*. Sydney.

McLaren, T. S., Head, M. M., Chan, Y.E. 2011. "A Multilevel Model for Measuring Fit Between a Firm's Competitive Strategies and Information Systems Capabilities." MIS Quarterly (35:4), pp. 909.

Mingers, J. 2004. "Realizing information systems: critical realism as an underpinning philosophy for information systems." Information and Organization (14:2), pp. 87-103.

Nunamaker Jr, J. F., Chen, M., Purdin, T.D.M. 1990. "Systems Development in Information Systems Research." Journal of Management Information Systems (7:3), pp. 89-106.

Orlikowski, W. J., Iacono, C.S. 2001. "Research commentary: desperately seeking the 'IT' in IT research-A call to theorizing the IT artifact." Information Systems Research (12:2), pp. 121-134.

Parsons, J., Wand, Y. 2008. "Using cognitive principles to guide classification in information system design." MIS Quarterly (32:4), pp. 839-868.

Peffers, K., Tuunanen, T., Rothernberger, M.A., Chatterjee, S. 2008. "A Design Science Research Methodology for Information Systems Research." Journal of Management Information Systems (24:3), pp. 45-78.

Popper, K. R. 2002. *Conjectures and refutations: The growth of scientific knowledge*, Routledge, NY.

Preece, J., Rogers, Y., Sharp, H. 2002. *Interaction design: beyond human-computer interaction*, John Wiley, UK.

Simon, H. A. 1996. *The sciences of the artificial*. Cambridge, MA, The MIT Press.

Siponen, M., Baskerville, R., Heikka, J. 2006. "A Design Theory for Secure Information Systems Design theorys TPF." Journal of the Association for Information Systems (7:11), pp. 725-770.

Straub, D.W., Gefen, D., Boudreau, M.C. 2005. "Quantitative Research", in *Research in Information Systems: A Handbook for Research Supervisors and Their Students*, D. Avison and J. Pries-Heje (Ed.), Elsevier, Amsterdam, pp. 221-238.

Takeda, H., Veerkamp, P., Tomiyama, T., Yoshikawam, H. 1990. "Modelling design processes." AI Magazine (11:4), pp. 37-48.

Tetzlajf, L., Schwartz, D.R. 1991. "The use of guidelines in interface design", in *Proceedings of the Conference on Human Factors in Computing Systems CHI'91*, ACM Press, NY, pp. 329-333

Truex, D., Holmström, J., Keil, M. 2006. "Theorizing in information systems research: A reflexive analysis of the adaptation of theory in information systems research." Journal of the Association for Information Systems (7:12), pp. 797-821.

Vaishnavi, V., Kuechler, W. 2007. *Design science research methods and patterns: innovating information and communication technology*. Auerbach Publications, NY.

Van Aken, J.E. 2005. "Management research as a design science: Articulating the research products of mode 2 knowledge production in management." British Journal of Management (16:1), pp. 19-36.

Venable, J. 2006. "The role of theory and theorising in Design Science research", *First International Conference on Design Science Research in Information Systems and Technology*. Claremont, California, pp. 1-18.

Vom Brocke, J., Buddendick, C. 2006. "Reusable Conceptual Models-Requirements Based on the Design Science Research Paradigm", in *Internation Conference on Design Science Research in Information Systems (DESRIST),* Claremont, CA, pp.576-604.

Walls, J.G., Widmeyer, G.R., El Sawy, O.A. 1992. "Building an information system design theory for vigilant EIS." Information Systems Research (3:1), pp. 36-59.

Walls, J.G., Widmeyer, G.R., El Sawy, O.A. 2004. "Assessing information system design theory in perspective: how useful was our 1992 initial rendition." Journal of Information Technology Theory and Application (6:2), pp. 43-58.

Weber, R. 2003. "Editor's comment: still desperately seeking the IT artifact." MIS Quarterly (27:2), pp. iii-xi.

Weick, K. E. 1995. "What Theory Is Not, Theorizing Is." Administrative Science Quarterly (40:3), pp. 385-390.

Winter, R. 2008. "Design science research in Europe." European Journal of Information Systems (17:5), pp. 470-475.

Yin, R. K. 2008. *Case study research: Design and methods*. London, Sage publications.

Zimmerman, J., J. Forlizzi, et al. (2007). *Research through design as a method for interaction design research in HCI*, ACM New York, NY, USA.