Chapter 5
Twelve Theses on Design Science Research in Information Systems

Some problems have such complex social, economic, or organizational interactions that they can't be solved fully. They've become popularly known as "wicked problems".

Robert W. Lucky, IEEE Spectrum, July 2009

This essay discusses 12 theses for guiding design science research. They are aimed at strengthening the design science orientation of Information Systems, clarifying future discourses on design science research aspects of the discipline, and giving some further guidelines for design science research in Information Systems.

5.1 Introduction

Although the current interest in design science research (DSR) (Nunamaker et al. 1990–1991; Walls et al. 1992; March and Smith 1995; Hevner et al. 2004; Gregor and Jones 2007) has been marked by an attempt to make it legitimate to do DSR in Information Systems (IS), DSR is still a sidetrack of IS research. Recognizing that IS ultimately is a practical discipline (Avison and Wood-Harper 1991), the message of the present chapter is that DSR should be its dominant research orientation. It is also important that the above articles have turned our attention to how to do rigorous DSR. Most notably, Hevner et al. (2004) propose seven guidelines for DSR and Gregor and Jones (2007) analyze the components of IS design theory.

Unfortunately, but understandably, the rapidly increased interest in DSR has led to uncertainty about what DSR is or should be (Baskerville 2008, Kuechler and Vaishnavi 2008, Winter 2008). In particular, its relation to "scientific design," "design science," and the "science of design" in the sense of Cross (1993, 2001) seems to be a source of continued confusion (McKay and Marshall 2007). The relationships between these and DSR will be elaborated at the end of the present chapter.

Juhani Iivari

A. Hevner, S. Chatterjee, *Design Research in Information Systems*, Integrated Series in Information Systems 22, DOI 10.1007/978-1-4419-5653-8_5,

The primary purpose of this chapter is to discuss 12 theses suggested in Iivari (2007) to summarize the disciplinary, ontological, epistemological and methodological analysis of IS as a design science. The 12 theses are as follows:

1. IS is ultimately an applied or practical discipline (discipline).
2. Prescriptive research is an essential part of IS as an applied or practical discipline (discipline).
3. The design science activity of building IT artifacts is an important part of prescriptive research in IS (discipline).
4. The primary interest of IS lies in IT applications, and therefore IS as a design science should be based on a sound ontology of IT artifacts and especially of IT applications (ontology).
5. IS as a design science builds IT meta-artifacts that support the development of concrete IT applications (ontology).
6. Prescriptive knowledge of IT artifacts forms a knowledge area of its own and cannot be reduced to the descriptive knowledge of theories and empirical regularities (epistemology).
7. The resulting IT meta-artifacts essentially entail design product and design process knowledge (epistemology).
8. The term "design theory" should be used only when it is based on a sound kernel theory (epistemology).
9. Constructive research methods should make the process of building IT meta-artifacts disciplined, rigorous, and transparent (methodology).
10. Explication of the practical problems to be solved, the existing artifacts to be improved, the analogies and metaphors to be used, and/or the kernel theories to be applied is significant in making the building process disciplined, rigorous, and transparent (methodology).
11. IS as a design science cannot be value-free, but it may reflect means-end, interpretive, or critical orientation (ethics).
12. The values of design science research should be made as explicit as possible (ethics).

These theses were not discussed in detail in Iivari (2007). The hope is that the following discussion will clarify the nature and role of DSR in IS and will give some further guidelines for such research.

5.2 Thesis 1: IS Is an Applied or Practical Discipline

There seems to be a certain reluctance in IS to characterize it as an applied discipline. One can identify two reasons for this. The first is that applied science may be deemed inferior to more "pure" science (Pitt 2000), and the second may be the conceptual confusion related to "applied science," "applied research," "pure science," and ”basic research.”[1] Referring to Strasser (1985), Gregor (2008) prefers the term “practical science,” and Avison and Wood-Harper (1991) characterize IS as a “practical discipline.”

It may be that terms such as “practical science” or “practical discipline” are more neutral than “applied discipline” for expressing the overall orientation of IS.[2] More essential than terminology, however, is the question of what implications this view has for IS research. Benbasat and Zmud (2003), for example, implicitly include the idea of IS as an applied or practical discipline in their statement of its aims:

> “our focus should be on how *to best design IT artifacts and IS systems* to increase their compatibility, usefulness, and ease of use or on how to best manage and support IT or IT-enabled business initiatives” [*italics* added by the author],

They nevertheless prefer to define the core of the field only in terms of a nomological net. As they do not recognize IS as a design science, their nomological net treats it as if it were only natural/behavioral research in which artifacts just happen to be part of the nomological net.

The characterization of IS as an applied or practical discipline strengthens its practical orientation: its general interest is in how to change the world and not only in how the world is. IS as an applied or practical discipline means that DSR is not a sidetrack, as is currently the situation, but should be its central orientation.

5.3 Thesis 2: Prescriptive Research Is an Essential Part of IS as an Applied or Practical Discipline

The idea of IS as an applied or practical discipline (Thesis 1) does not mean that it should include only “applied research.” Most disciplines comprise both “basic research” and “applied research.” When speaking about various types of research within a discipline, I find the distinction between descriptive and prescriptive research clearer than that between “basic research” and “applied research.”

Bazerman (2005) recommends that social sciences should have more prescriptive implications for organizations and for society at large, claiming that economics has been more successful in deriving theoretical implications than the other social sciences. Indeed, economics provides a good example of descriptive and prescriptive research. Adapting Chmielewicz (1970), Lehtovuori (1973) proposes that one can identify four levels in economics as a discipline: the conceptual level, the descriptive level of economic theory, the prescriptive level of economic policy, and the normative level of economic philosophy. The research goal at the conceptual level

[1]Referring to the first reason, the “anxiety discourse” (King and Lyytinen 2004) regarding the academic legitimacy and credibility of the discipline has been an amazingly significant issue in information systems, guiding far too much of the evolution of the discipline.

[2]Hassan (2006) points out that it is more appropriate to speak about Information Systems as a field than as a discipline. The reasons are its lack of theory development and its weak boundaries. Despite this inaccuracy, I will speak below about the “IS discipline”.

is essentialist: concepts and conceptual frameworks do not have any truth value or "truthlikeness" (Niiniluoto 1999), but simply attempt to capture the essence of the phenomena. The research goal at the level of economic theory is theoretical, to find causal relationships, and that at the level of economic policy is pragmatic, to find means-end relationships. Both causal and means-end relationships have a truth value. The level of economic philosophy has a normative research goal, being concerned with values that do not have any truth value.

The resultant structure when applied to IS is illustrated in Fig. 5.1.[3] Concepts and conceptual frameworks at the conceptual level aim at identifying essences in the research territory and their relationships. They may be more or less useful when developing theories at the descriptive level, which aim at describing, understanding and explaining how things are.

Stated briefly, the conceptual level is interested in "what things are out there," descriptive research in "how things are out there," and prescriptive research in "how things could be out there" and "how one can effectively achieve specified ends". The prescriptive level covers both recommendations and artifacts as outcomes of DSR. These do not have any truth or truth-like value as such, but statements about their efficiency and effectiveness do.

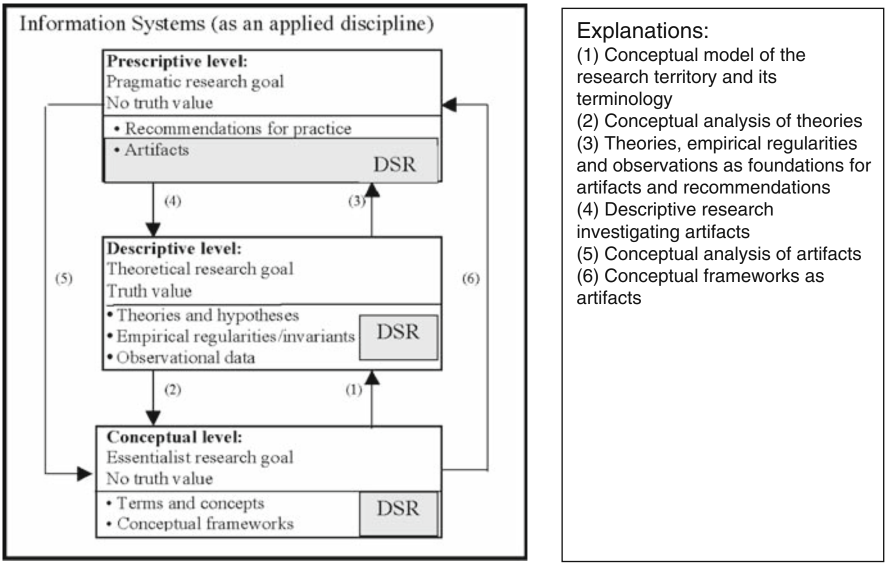

Fig. 5.1 Three levels of research in information systems

[3]Figure 5.1 drops the normative level of the original framework of Chmielewicz (1970) and Lehtovuori (1973). The normative level is interested in "how ought things to be?" Normative statements express "You ought to want A and to achieve this you should do X if you believe that you are in a situation B." The reason for the exclusion is that it is still a controversial question whether one can reach "ought-to" conclusions based on "what is."

The hierarchy of Fig. 5.1 can be mapped to the types of "theories" suggested by Gregor (2006). "Theories for analyzing and describing" lie at the conceptual level, "theories for predicting" are empirical regularities, "theories for explaining and predicting" refer to theories at the descriptive level, and "theories for design and action" represent the prescriptive level. Only "theories of explaining," when interpreted as grand theories such as critical social theory, structuration theory, actor-network theory, activity theory, do not have any representation in Fig. 5.1.

5.4 Thesis 3: The Design Science Activity of Building IT Artifacts Is an Important Part of Prescriptive Research in Information Systems

Figure 5.1 also illustrates the position of DSR in the framework, indicating that it may be conceptual, descriptive or prescriptive. [4] Philosophical treatments of prescriptive research (Bunge 1967b; Niiniluoto 1993) tend to interpret the prescriptive level as comprising only prescriptions based on practical implications of descriptive research and do not recognize complex artifacts as research outcomes. Niiniluoto (1993), for example, suggests that the typical knowledge claims of descriptive science are deterministic or probabilistic causal laws such as

(1) X causes A in situation B
(2) X tends to cause A in situation B with probability *p*

One can also derive predictions from these descriptive laws:

(3) X causes A in situation B
X occurred in situation b
The situation b is of type B
Hence, A will occur in b

and also technical norms such as

(4) If you want A and you believe that you are in a situation B, then

- you should do X (if X is a necessary cause of A)
- it is rational for you to do X (if X is a sufficient cause of A)
- it is profitable for you to do X (if X is a probabilistic cause of A)

[4] Interestingly, Winter (2008) applies the tenets of a 1990 edition of Chmielewicz's book (underlying Fig. 5.1) to structure DSR. His mapping of models, methods, constructs, and theories is quite consistent with Fig. 5.1, but he associates instantiations with the normative level. This differs from my interpretation of Chmielewicz (1970) based on Lehtovuori (1974).

Niiniluoto (1993) gives some examples of X, such as medical treatments, fertilizers, and materials used in aeroplanes, but fails to explicitly recognize X's as results of DSR or design product knowledge of X's as a separate category of knowledge at the prescriptive level.

The claim of thesis 3 is that the DSR activity of building IT artifacts is an important part of prescriptive research in IS. Evaluation as a DSR activity lies at the descriptive level. It studies how effective and efficient the artifacts are compared with existing artifacts. As illustrated by Hevner et al. (2004), evaluation applies the very same research methods as does descriptive research more generally. As such, descriptive DSR (i.e., evaluation) does not differ much from other descriptive research. In fact, if the plea of Orlikowski and Iacono (2001) to take the IT artifact seriously in IS research is to be heeded, much of it could be descriptive DSR, making the borderline between evaluation as a DSR activity and more general descriptive research increasingly diffuse.

As pointed out by March and Smith (1995), many artifacts are primarily concepts (constructs) or conceptual frameworks (models and methods). Therefore the building of constructs, models, and methods is indicated in Fig. 5.1 as both prescriptive and conceptual research at the same time.

5.5 Thesis 4: The Primary Interest of IS Lies in IT Applications, and Therefore IS as a Design Science Should Be Based on a Sound Ontology of IT Artifacts and Especially of IT Applications

The three worlds of Popper (1978) provide a good starting point for such an ontology (Iivari 2007). World 1 is about material nature, World 2 about consciousness and mental states, and World 3 about products of human social action. World 3 clearly includes human artifacts, and it also covers institutions and theories, where institutions are social constructions that have been objectified (Berger and Luckman 1967).

If we conceive of IS as a design science that also builds IT artifacts, a natural question is what sort of artifacts we build, especially if we wish to distinguish IS from its sister disciplines, computer science and software engineering, which also focus on IT artifacts. I would suggest that the primary interest of IS lies in IT applications, and therefore IS as a design science should be based on a sound ontology of IT artifacts and especially of IT applications.

The typology for IT applications proposed in Iivari (2007) distinguishes seven archetypes of IT applications based on the function/role that the application serves: automating, augmenting, mediating, informing, entertaining, artisticizing, and accompanying. One could add fantasizing applications to this list. The first four functions are close to "technology as a labor substitution tool," "technology as a productivity tool," "technology as a social relations tool," and "technology as an information processing tool" in Orlikowski and Iacono (2001). Thus the typology essentially extends these four categories by incorporating four additional ones. Computer games illustrate the capability of IT applications to entertain. IT applications may also attempt to arouse artistic experience, and one can easily imagine a new sort of art that is essentially built on the interactive character of computer technology. IT artifacts such as digital pets can accompany human users. Finally, virtual fantasy worlds such as Second Life allow fantasizing applications.

A sound typology of IT artifacts, and especially of IT applications, is significant for a number of reasons. First, it is obvious that IT artifacts differ in design. A compiler design, for example, is quite different from the design of a specific information system, and the designing of an information system differs from game design. Second, as Swanson (1994) and Lyytinen and Rose (2003) suggest, IT artifacts differ in their diffusion. Third, it is my conjecture that IT application archetypes also differ in their acceptance, so that the Technology Acceptance Model (Davis et al. 1989; Venkatesh et al. 2003) is valid only in the case of certain IT application artifacts.

5.6 Thesis 5: IS as a Design Science Builds IT Meta-artifacts That Support the Development of Concrete IT Applications

One should note, however, that IS as a design science does not attempt to develop concrete IT applications, but rather meta-artifacts that help develop the concrete IT applications. van Aken (2004) makes a similar distinction between general solution concepts (meta-IT artifacts) and specific solution concepts (concrete IT applications).

Making a similar distinction, Walls et al. (1992) speak about meta-requirements and meta-design. Meta-artifacts can further be divided into *meta-artifacts for the IT product* and *meta-artifacts for the systems development process*. In the case of information systems, the former comprise technical implementation resources such as application domain-specific software components, application frameworks, application packages, ERP systems, development environments, IS generators, or their prototypes, which can be used in the technical implementation of an IS artifact, and also more abstract models and principles such as IS meta-models, various architectural models, analysis and design patterns and application-dependent design principles for use in the design and implementation of the IS product, while the latter correspond to the "design process" in the information system design theory of Walls et al. (1992) and comprise systems development approaches, methods, techniques and tools, for example.

Kuechler and Vaishnavi (2008) criticize the constructivist view of DSR adopted in Thesis 5 that emphasizes artifacts as research outputs of DSR. One should note, however, that the distinction between DSR and descriptive (behavioral) research is first of all epistemological. Descriptive research attempts to produce empirical regularities and theoretical understanding that can be assessed in terms of truth or "truthlikeness," whereas artifacts as outputs of DSR are only useful to varying extents. Hevner et al. (2004) clearly recognize this epistemological difference when they state that the goal of behavioral science research is truth and the goal of DSR is utility (p. 80).

5.7 Thesis 6: Prescriptive Knowledge of IT Artifacts Forms a Knowledge Area of Its Own and Cannot Be Reduced to the Descriptive Knowledge of Theories and Empirical Regularities

Niiniluoto's (1993) technical norms (see Thesis 3) give an impression that design science knowledge (technical norms) is largely reducible to descriptive knowledge (causal laws). The relationship between science and technology has been of considerable interest (Gardner 1994, 1995), leading to the conclusion that descriptive science and technology are separate, even though mutually interacting, bodies of thought, and that prescriptive knowledge cannot be reduced to descriptive knowledge of theories and empirical regularities (Layton 1974). [5]

The link between descriptive research and prescriptive research seems to be particularly weak in IS, where IT artifacts are relatively independent of descriptive theories concerning nature, human beings, organizations and other institutions, although quite recently IT has enabled new organizational forms to be developed based on networking and virtuality. Even though technical implementability is a significant issue, the dependence of IT artifacts on the laws of nature is mainly latent, and IS designers do not need to be constantly considering them. One can expect that the need for theories of human beings is the most obvious in the context of human–computer interaction (HCI), but the theoretical foundation of HCI is unclear and fragmented (Clemmensen 2006). It is also uncertain to what extent existing theories inform HCI design either directly or indirectly through design methods, standards, guidelines, etc.

The situation in IS is very similar. It has a diversity of reference disciplines from which it has adopted a number of theories (Benbasat and Weber 1996), but these theories are weakly linked to IT artifacts and their design. Even so, people design reasonably successful IT artifacts. This makes one to wonder whether the IS research community tends to exaggerate the significance of descriptive theoretical knowledge for prescriptive knowledge regarding how to design successful IT artifacts.

[5] Lyytinen and King (2004) also touch upon this issue when criticizing the linear science -> technology -> society model. One should note, however, that they do not go very far in their criticism when discussing the cyclical society -> science -> technology -> society model as an alternative.

5.8 Thesis 7: The Resulting IT Meta-artifacts Essentially Entail Design Product and Design Process Knowledge

Bunge (1967a) notes that the primary target of any scientific research, whether pure or applied, is to advance knowledge. Pure science has a purely cognitive aim, whereas applied science (technology) also has practical, utilitarian aims. If we accept Bunge's view and take IT meta-artifacts seriously as major outputs of DSR, this will imply that meta-artifacts for design product and systems development process (see Thesis 5) essentially entail knowledge. This is in line with Walls et al. (1992), who conceptualize meta-artifacts as design theories, and Hevner et al. (2004), who include IT artifacts (constructs, models, methods, and even instantiations) in the knowledge base.

van Aken (2004) claims that "the mission of a design science is to produce knowledge for the design and realization of artifacts, i.e. to solve construction problems, or to be used in the improvement of the performance of existing entities, i.e. to solve improvement problems." He also suggests three types of design science knowledge: object knowledge of the characteristics of artifacts and their materials, realization knowledge of the physical processes to be used to realize the designed artifacts, and process knowledge, of characteristics of the design process (van Aken 2005). In addition, he links the three types of design knowledge to technical norms of the form "if you wish to achieve A in situation B, then do something like X."[6] X in technical norms may refer to object design, realization design, or process design.

The distinction between design product knowledge, technological rules, and technical norms in prescriptive design science knowledge is set out in Table 5.1. The design product knowledge embedded in artifacts is a relatively weakly understood form of knowledge. The first three aspects of design product knowledge in Table 5.1 are close to the three criteria for artifacts identified by Beckman (2002): intentional, operational, and structural. Beckman illustrates these in the case of "knifehood." The intentional criterion implies that a thing is a knife because it is used as a knife, the operational criterion means that a thing is a knife because it works like a knife, and the structural criterion suggests that a thing is a knife because it has the shape and fabric of a knife. Beckman (2002) also includes a fourth criterion, the conventional one, which implies that a thing is a knife because it fits the reference of the common concept of "knife." In the DSR context, the conventional criterion is a significant goal in the sense that the artifact (e.g., a new systems development method OO+++) will be accepted as a valid instance of a given class concept (e.g., object-oriented methods) by a relevant community (e.g., practitioners). Despite this, I do not think that it is an inherent aspect of the artifact, since the artifact may achieve

[6] van Aken is referring here to technological rules (Bunge 1967b) of the following type: in order to achieve A do acts 1–*n* in a given order. One can interpret technological rules in the sense of Bunge (1967b) as expressing design process knowledge, but van Aken interprets them as technical norms in the sense of Niiniluoto (1993).

Table 5.1 Prescriptive design science knowledge

Design product knowledge	The artifact – idea, concept, style – functionality, behavior – architecture, structure – possible instantiation
Design process knowledge Technological rules (Bunge 1967b) Technical norms (Niiniluoto 1993)	In order to achieve A – do (act_1, act_2, . . ., act_n) If you want A and you believe that you are in a situation B, then – you should do X – it is rational for you to do X – it is profitable for you to do X

general community acceptance years after its invention and construction. Therefore, the conventional criterion is not explicitly listed in Table 5.1, but following March and Smith (1995), instantiation is included as a fourth aspect.

It should be noted that some DSR literature tends to emphasize the significance of instantiations as research outcomes of DSR. Instantiations are, of course, significant as "proofs of a concept" (Nunamaker et al. 1990–1991). They may also increase the practical utility of the ideas, but from the research point of view they are secondary. The essential thing is the design product knowledge they entail.

5.9 Thesis 8: The Term "Design Theory" Should Be Used Only When It Is Based on a Sound Kernel Theory

Walls et al. (1992) pioneered the idea that design science should be rooted in theories. Ideally, theories should serve as sources of ideas in DSR, and they suggested that an "IS design theory" for a product should consist of meta-requirements (the class of goals to which the theory applies), meta-design (the class of artifacts hypothesized to meet the meta-requirements), kernel theories (theories from the natural and social sciences governing design), and testable design product hypotheses (used to test whether the meta-design satisfies the meta-requirements). An "IS design theory" for a process would comprise a design method (a description of the procedures for artifact construction), kernel theories, and testable design process hypotheses (used to verify whether the design method results in an artifact which is consistent with the meta-design).

Although I am afraid that the strong theory orientation of the leading IS journals may exaggerate the dependence of prescriptive knowledge on descriptive knowledge (see Thesis 6), I would consider the existence of a kernel theory to be a defining characteristic of a "design theory." Since Walls et al. (1992) point out that kernel theories are derived from the natural and social sciences and from mathematics, I wish to point out that it is not necessary for a kernel to be from some reference discipline external to IS. A kernel theory can be a theory specific to IS. As stated by Gregor (2006), a kernel theory may be a descriptive IS-specific "theory for predicting" or "explaining and predicting," an IS-specific theory "for analyzing and describing," or even another IS design theory or "theory for design and action," provided that the kernel theory is considered sound enough by the relevant scientific community.

Essentially, the claim is that without a sound kernel theory it is not justified to speak about "design theory." This is quite an ambitious requirement, because it is difficult, as Walls et al. (1992) demonstrate, to find convincing examples of IT meta-artifacts with well-defined kernel theories. As a result there seems to be some tendency to soften the requirements for a kernel theory. Markus et al. (2002), for example, allow any practitioner theory-in-use to serve as a kernel theory. This implies that a design theory is not necessarily based on any scientifically validated knowledge. Taking a cynical viewpoint, if kernel theory is forgotten, there is a danger that the idea of a "design theory" will be (mis)used just to make our field sound more scientific without any serious attempt to strengthen the scientific foundation of the meta-artifacts proposed. [7]

5.10 Thesis 9: Constructive Research Methods Should Make the Process of Building IT Meta-artifacts Disciplined, Rigorous, and Transparent

Recognizing that much of the research in computer science and software engineering in particular has consisted of constructing artifacts, the term "constructive research" was suggested in Iivari (1991) to denote the specific research methods required for constructing artifacts. [8] Although well-recognized in the design science literature, the building of artifacts is relatively poorly understood as a design science research activity, especially as compared with evaluation. [9] March and Smith (1995) do not have much to say about the activity of constructing artifacts, although they do point out the novelty of an artifact (construct, model, method, or instantiation) and the persuasiveness of the claims that the new artifact should be effective. They also emphasize that instantiations that apply known constructs, models, and methods to

[7] In fact, I think that Walls et al. (1992) fall into this trap when they suggest that the information systems development life cycle is a design theory. I am not aware of any kernel theory on which it is based.

[8] Note that well-known classifications of IS research methods such as those of Benbasat (1985), Jenkins (1985), and Galliers and Land (1987) do not recognize anything resembling constructive research methods nor, even, does a recent review of research methods in the IS literature (Chen and Hirschheim, 2004).

[9] The article of Hevner et al. (2004) illustrates this. They suggest a detailed list of methods for evaluation, but nothing corresponding to the building of artifacts. There is also a rich body of literature on evaluation that can be applied in the design science context (Verschuren and Hartog 2005).

novel tasks may be of little significance if there is not sufficient uncertainty about their applicability.

The seven design science research guidelines suggested by Hevner et al. (2004) do not directly address the question of how artifacts are built, although many of them touch upon the topic. Guideline 1 suggests artifacts as products of design science research, Guideline 2 emphasizes that design science research should develop technology-based solutions to important and relevant problems, and Guideline 4 discusses the contributions of design science research, emphasizing that the artifact must be innovative, solving a heretofore unsolved problem or solving a known problem in a more effective or efficient manner (p. 82). The novelty of artifacts makes it possible to distinguish IS from the ordinary practice of developing IT artifacts. Guideline 5, concerning research rigor, imposes a requirement that design science research must apply rigorous methods for both the construction and evaluation of artifacts. This rigor, according to Hevner et al. (2004), should be derived from the effective use of *prior* research (the existing knowledge base). Guideline 6 suggests that design is essentially a search process for discovering an effective solution to a problem, largely following Simon (1969/1981/1996) in this respect. I find this idea of the building of artifacts as problem solving somewhat problematic, for two reasons. First, what the problem is is often a problem. The problem is not necessarily given, but instead the researcher has considerable discretion in deciding what the problem is. Thus the constructing of a design science artifact is as much problem setting as problem solving. Second, design as a search process implies an idea that alternatives are there to be discovered. In reality they are not, but rather they must be constructed in some way. [10]

To my knowledge, treatment of how to build artifacts in DSR provided by Nunamaker et al. (1990–1991) is still the most refined of its type. They propose that systems development could serve as a specific research method for constructing artifacts, introducing a model of four interacting research activities, theory building, experimentation, observation, and systems development, where systems development lies at the center. The process that they propose for systems development is quite a conventional software development model. In as far as the artifacts to be built are systems, systems development is a natural candidate for methods of constructive research. The method seems particularly relevant when the purpose is to validate the concept by implementing (instantiating) the system. One should note, however, that not all artifacts developed in DSR within computer science, information systems, and software engineering are information or software systems (e.g., systems development methods), and it is an open question as to what extent systems development methods work as research methods. If systems development methods really are applicable, this should put an end to the regression of meta-levels between artifacts, since systems development methods, as meta-artifacts for the IS development process, could be employed for developing other meta-artifacts.

[10] Despite of these critical comments, I see problem solving as a useful heuristic metaphor to be used when considering alternative solutions, especially for different components of the artifact.

It is widely understood that the building of artifacts in DSR is at least ideally a creative process (Nunamaker et al., 1990–1991; March and Smith 1995; Hevner et al. 2004). One could maintain that it has a lot in common with theory building, which has been of interest in the methodology of science (e.g., Dubin 1969). One can speculate, however, that artifacts in particular leave much more space for creative imagination, since they are not assumed to describe or explain any existing reality. IT artifacts may create their own virtual world (e.g., computer games, computer art, computer pets, and virtual fantasy world applications) in which the laws of nature, for example, are not valid. Because of the creative element, it is difficult to define an appropriate method for the design science activity of artifact building.

Despite the above difficulty, I see the existence of constructive research methods as highly essential to the identity of IS as a design science. It is the rigor of constructing IT artifacts that distinguishes IS as a design science from the practice of building IT artifacts. One should note here that the construction of innovative IT artifacts (or IT meta-artifacts) is not a monopoly of the research community, but practitioners may also do it. Acknowledging this, there are two options for demarcating IS as a design science from inventions made by practitioners. The first is to accept that there is no constructive research method that distinguishes the two, but that the difference lies in the evaluation: the essence of IS as a design science lies in the scientific evaluation of artifacts. This is one option, but it easily leads to reactive research in which IS as a design science focuses on the evaluation of existing IT artifacts rather than on the building of new ones.

The second option is to try to specify a reasonably rigorous constructive research method for building IT artifacts. It would then be this method that differentiated the design science construction of IT artifacts from the Gyro Gearloose style of invention in practice. [11] If a practitioner applies the same rigor as an IS researcher, he/she is essentially a researcher. I would expect that this would make IS as a design science more proactive, attempting to guide the evolution of IT and not merely react to it.

5.11 Thesis 10: Explication of the Practical Problems to Be Solved, the Existing Artifacts to Be Improved, the Analogies and Metaphors to Be Used, and/or the Kernel Theories to Be Applied Is Significant in Making the Building Process Disciplined, Rigorous, and Transparent

Should an artifact as an outcome of DSR always be based on recognizable theory? March and Smith (1995) point out that design science artifacts are often invented without any clear descriptive theory. The possibility of an IT artifact not having any

[11] Gyro Gearloose is a fictional character created by Carl Barks for the Walt Disney Company. The purpose of using this figure to symbolize inventors in the field is not to ridicule them, but quite the contrary.

kernel theory raises the question of the criteria governing whether an artifact can be considered a scientific contribution and publishable in IS journals. The requirement of an underlying descriptive theory may considerably limit DSR, possibly excluding the most innovative design science outcomes from major IS journals. [12] As noted above, Guideline 4 in Hevner et al. (2004), that the artifact must be innovative, solving a heretofore unsolved problem or solving a known problem in a more effective or efficient manner leads to an additional question of whether complete evaluation of the proposed artifact is required. The situation is analogous to theory building: if the building of a theory is accepted as a scientific contribution without complete testing, why cannot the building of a novel IT meta-artifact also be accepted without complete evaluation, provided that the IT meta-artifact is novel and well-argued? The idea of an IT meta-artifact being well-argued means that it cannot come "out of the blue," but must be rigorously constructed from specific origins.

Hevner et al. (2004) propose that the rigor of DSR should be derived from the effective use of *prior* research (an existing knowledge base). I would claim that the construction process should also be made as transparent as possible if it is to be considered a design science activity. Knowing that these proposals are preliminary, I suggest four major sources of ideas for DSR to make the building process more disciplined, rigorous, and transparent:

1 Practical problems and opportunities
2 Existing artifacts
3 Analogies and metaphors
4 Theories

The first of these emphasizes the practical relevance of research. Furthermore, it is well known in innovation diffusion research (Rogers 1995) that customers serve as a significant source of innovations (von Hippel 1988), especially in the case of IT innovations (von Hippel 2005). I do not claim that researchers should attempt to solve practical problems exactly as they appear in practice. A practical problem may be a conglomerate of different problems, and a piece of research may not attempt to address the whole conglomerate but may focus only on a specific subproblem. A practical problem may also be abstracted to make it more general and easier to link to theories. One should note, however, that design science is also about potentiality. A new idea or artifact may provide totally new opportunities to improve practice long before practitioners recognize any problem. There are many significant innovations in our field that illustrate this, such as the relational data model and the first ideas of object orientation.

Most DSR consists of incremental improvements to existing artifacts, as illustrated by research into conceptual information modeling in the 1970s and into

[12] Could Berners-Lee, for example, have published his ideas on WWW in a top IS journal?

object-oriented systems development in the 1990s. Typically, the marginal value of additional improvements decreases until the research gradually fades out. [13]

It is also well known that analogies and metaphors stimulate creativity (Couger et al. 1993). In the case of IT artifacts, for instance, cognitive and biological theories have provided useful metaphors for computing, such as neural networks and genetic algorithms. The desktop metaphor led to the graphical user interfaces which predominate nowadays, and the spreadsheet metaphor led to spreadsheet software, which forms one of the most widely applied personal productivity tools.

5.12 Thesis 11: IS as a Design Science Cannot Be Value-Free, but It May Reflect Means-End, Interpretive, or Critical Orientation

DSR in itself implies an ethical change from describing and explaining the existing world to shaping it. The ethics of research concern the responsibility of a scientist for the consequences of his research and its results. Even though it may be questionable whether any research can be value-free, it is absolutely clear that DSR cannot be. Consequently, the basic values of research should be expressed as explicitly as possible.

Adapting Chua (1986), Iivari (1991) distinguished three potential roles for IS as an applied discipline: (1) means-end oriented, (2) interpretive, and (3) critical. In the first case the scientist aims at providing knowledge as a means for achieving given ends (goals), without questioning the legitimacy of those ends. According to Chua (1986), the aim of an "interpretivist scientist is to enrich people's understanding of their action", "how social order is produced and reproduced" (p. 615). The goals (ends) of action are often not so clear, and one should also focus on unintended consequences. A critical scientist will see that research has "a critical imperative: the identification and removal of domination and ideological practice" (p. 622). Goals (ends) can be subjected to critical analysis. [14]

Much DSR is naturally means-end oriented. This concerns especially constructive research involved with the building of artifacts. But constructive research can also be critical, as exemplified by the Scandinavian trade-unionist systems development approach (Bjerknes et al. 1987). Evaluation studies can be means-end oriented, interpretive, and/or critical, where a means-end-oriented evaluation is only interested in how effectively the artifact helps achieve the given goals or ends, an interpretive piece of evaluation research may attempt to achieve a rich understanding

[13] One can, of course, observe a similar phenomenon in descriptive research, as illustrated by the extensions of the Technology Acceptance Model (Davis et al. 1989).

[14] Note that Iivari (1991) applied the above distinction as an ethical dimension, whereas Orlikowski and Baroudi (1991) applied a very similar distinction as an epistemological dimension. The critical perspective clearly illustrates the problem with the epistemological dimension. Critical research may apply either a positivistic or an anti-positivistic epistemology.

of how an IT artifact is really appropriated and used and what its effects are, without confining the focus on the given ends of its initial construction; and a critical study is interested in how an IT artifact enforces or removes unjustified domination or ideological practices.

5.13 Thesis 12: The Values of Design Science Research Should Be Made as Explicit as Possible

More concretely, one can also question the values of IS research, i.e., whose values and what values dominate it, emphasizing that research may openly or latently serve the interests of particular dominant groups. The interests served may be those of the host organization as perceived by its top management, those of IS users, those of IS professionals, or potentially those of other stakeholder groups in society.

5.14 Conclusions and Final Comments

The aim of this chapter is to strengthen the design science orientation of IS. If fully adopted, this orientation would mean profound changes in the disciplinary identity of IS, in its ontology, epistemology, methodology, and ethics. It will not necessarily be easy to get these changes understood and accepted in the IS research community. In addition to natural resistance to change, there is a certain ambiguity in the idea of design science research.

The idea of DSR in IS is still in its formative stage. As new members join the DSR community, each of them may bring in his or her own interpretation of what DSR is. While the plurality of ideas is definitely beneficial, especially at this early stage, it is also good for people to understand what they are talking about. Individual keywords in DSR such as "design" and "artifact" can easily be misleading, since IS development in practice is essentially design, and the concept of "artifact" can be interpreted very broadly to cover all World 3 objects and phenomena in the ontology of Popper (1978). The phrase "design science" is also problematic, since it is used in a quite different meaning in the design studies community (Cross 1993, 2001) from that used by Walls et al. (1992), March and Smith (1995), and Hevner et al. (2004), where the focus is clearly on DSR.

Regarding the attempt to clarify the relationship between DSR and related research areas as set out in Fig. 5.2, one should note that articles such as Nunamaker et al. (1990–1991), March and Smith (1995), Hevner et al. (2004), and the present chapter represent research into DSR in IS. In addition to advocating the need for DSR research in IS, they attempt to provide concepts (March and Smith 1995), principles (Hevner et al. 2004), theses (the present work), and research methods (Nunamaker et al. 1990–1991) for DSR in an IS context.

Essentially following Walls et al. (1992), DSR in IS is divided in Fig. 5.2 into DSR focused on IT products and DSR focused on the systems development process.

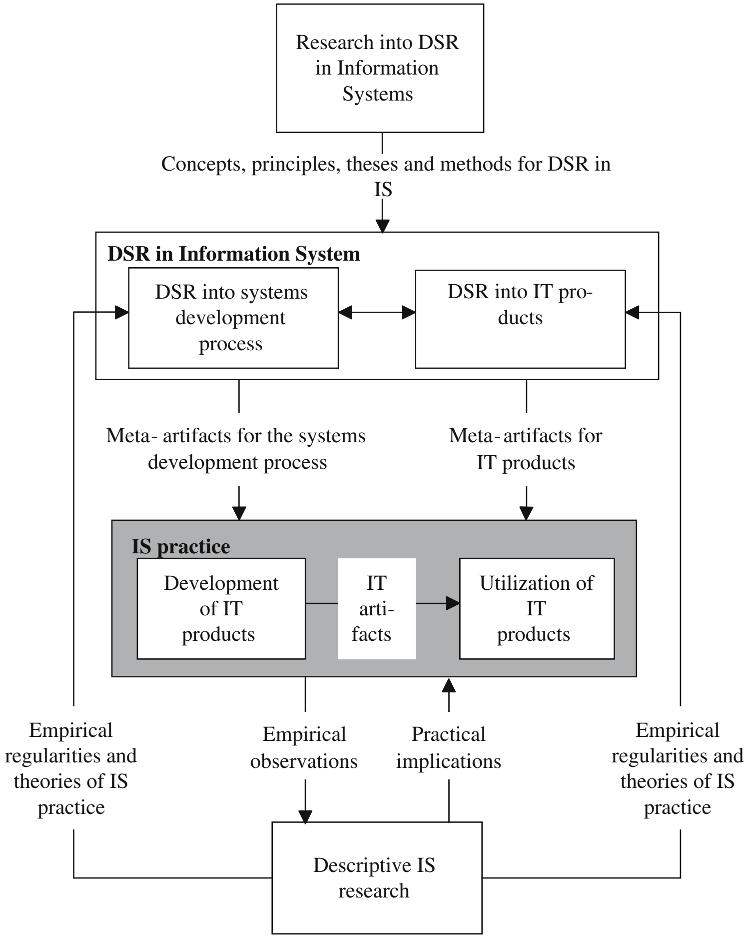

Fig. 5.2 A framework for design-related research areas in information systems

These are assumed to produce meta-artifacts for IT products and meta-artifacts for the systems development process for use in IS practice (see Thesis 5 above), where IS practice is taken to comprise the development of IT products (especially concrete IT applications) to be utilized in practice.

The IS practice of developing and utilizing IT products forms a central phenomenon to be investigated and understood by descriptive IS research. This descriptive research may provide a scientifically founded understanding of IS practice that may support DSR in IS, and it may also convert the observed empirical regularities and validated theories into prescriptive practical implications (recommendations).

Figure 5.2 makes it possible to understand the relationships between DSR and "scientific design," "design science," and the "science of design" as characterized by Cross (1993, 2001). According to Cross (2001, p. 53), "scientific design" means that design products should be based on scientific knowledge, "design science" means that the design process is based on "an explicitly, organized, rational, and wholly systematic approach to design", as if the design process was "a scientific activity in itself", and the "science of design" means the scientific study of design activity itself. If confined to IT products, "scientific design" resembles DSR into IT products in Fig. 5.2, although the latter does not necessarily share the idea that design products should be based on scientific knowledge (cf. Thesis 6 above). Similarly, "design science" resembles DSR into the systems development process, although not all DSR into the systems development process shares the idea that the design process should be based on an explicitly, organized, rational, and wholly systematic approach to design. Descriptive IS research includes the "science of design" when the focus of the latter lies in the design of IT artifacts. Note also that DSR includes design, which may also be the focus of "science of design." [15]

As pointed out above, the work of Hevner et al. (2004) does not represent DSR in Information Systems, but is research into DSR in Information Systems. Although their work has been criticized for adopting a biased and narrow view of information systems and IS design (McKay et al. 2008), its greatest weakness in my view is its generality. If the IS/IT-specific examples are deleted and the IS/IT-specific terms are translated into more neutral ones such as "design" and "artifact," their framework for DSR is a very general one and not particularly specific to Information Systems. One challenge for future research, to my mind, will be to refine Hevner et al. (2004) toward a more IS/IT-specific version.

References

Avison, D. E. and A. T. Wood-Harper (1991) Information systems development research: an exploration of ideas in practice, *The Computer Journal* 34 (2), pp. 98–112.

Baskerville, R. (2008) What design science is not, *European Journal of Information Systems* 17, pp. 441–443.

Bazerman, M. H. (2005) Conducting influential research: the need for prescriptive implications, *Academy of Management Review* 30 (1), pp. 25–31.

Beckman, S. (2002) The nature of artifacts, in Dahlbom, B., Beckman, S. and Nilsson, G. B., *Artifacts and Artificial Science*, Almqvist & Wiksell International, Stockholm, pp. 45–92.

Benbasat, I. (1985) An analysis of research methodologies, in *The Information Systems Research Challenge*, McFarlan, F. W. ed., Harvard Business School Press, Boston, MA, pp. 47–85.

Benbasat, I. and R. Weber (1996) Rethinking diversity in information systems research, *Information Systems Research* 7 (4), pp. 389–399.

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

Berger, P. and T. Luckman (1967) *The Social Construction of Reality: A Treatise on Sociology of Knowledge*, Doubleday, New York, NY.

Bjerknes, G., P. Ehn, and M. Kyng (eds.) (1987) *Computers and Democracy*, Avebury, Aldershot.

[15] Hevner et al. (2004) characterize the difference between design activities by stating that design in IS practice is routine and design as part of DSR is more innovative. This is a slightly unfortunate characterization, since design in IS practice is frequently anything but routine.

Bunge, M. (1967a) *Scientific Research I, The Search for System*, Springer-Verlag, New York.

Bunge, M. (1967b) *Scientific Research II. The Search for Truth*, Springer-Verlag, New York.

Chen, W. and R. Hirschheim (2004) A paradigmatic and methodological examination of information systems research from 1991 to 2001, *Information Systems Journal* 14 (3), pp. 197–235.

Chmielewicz, K. (1970) *Forschungskonzeptionen der Wirtschaftswissenschaft*, Stuttgart.

Chua, W. F. (1986) Radical developments in accounting thought, *The Accounting Review*, LXI (5), pp. 583–598.

Clemmensen, T. (2006) Whatever happened to the psychology of human–computer interaction? A biography of the life of a psychological framework within a HCI journal, *Information Technology & People* 19 (2), pp. 121–151.

Couger, J. D., L. F. Higgins, and S. C. Mcintyre (1993) (Un)structured creativity in information systems organizations, *MIS Quarterly* 17 (4), pp. 375–397.

Cross, N. (1993) Science and design methodology, *Research in Engineering Design* 5, pp. 63–69.

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

Davis, F. D., R. P. Bagozzi, and P. R. Warshaw (1989) User acceptance of computer technology: a comparison of two theoretical models, *Management Science* 35 (8), pp. 982–1003.

Dubin, R. (1969) *Theory Building*, The Free Press, New York.

Galliers, R. D. and F. F. Land (1987) Choosing appropriate information systems research methodologies, *Communications of the ACM* 30 (11), pp. 900–902.

Gardner, P. L. (1994) The relationship between technology and science: some historical and philosophical reflections. Part I, *International Journal of Technology and Design Education* 4 (2), pp. 123–153.

Gardner, P. L. (1995) The relationship between technology and science: some historical and philosophical reflections. Part II, *International Journal of Technology and Design Education* 5 (1), pp. 1–33.

Gregor, S. (2006) The nature of theory in information systems, *MIS Quarterly* 30 (3), pp. 611–642.

Gregor, S. (2008) Building theory in a practical science, in *Information Systems Foundations: Answering the Unanswered Questions About Design Research, The 4th ANU Workshop on Information Systems Foundations*, Australian National University, Canberra, Australia.

Gregor, S. and D. Jones (2007) The anatomy of a design theory, *Journal of the AIS* 8 (5), pp. 312–335.

Hassan, N. R. (2006) Is information systems a discipline? A Foucauldian and Toulminian analysis, in *Proceedings of the Twenty-Seventh International Conference on Information Systems*, Milwaukee, pp. 425–440.

Hevner, A. R., S. T. March, J. Park, and S. Ram (2004) Design science in information systems research, *MIS Quarterly* 28 (1), pp. 75–105.

Iivari, J. (1991) A paradigmatic analysis of contemporary schools of IS development, *European Journal of Information Systems* 1 (4), pp. 249–272.

Iivari, J. (2007) Paradigmatic analysis of information systems as a design science, *Scandinavian Journal of Information Systems* 19 (2), pp. 39–63.

Jenkins, A. M. (1985) Research methodologies and MIS research, in Mumford, E., Hirschheim, R., Fitzgerald, G. and Wood-Harper, A.T. (eds.), *Research Methods in Information Systems,* North-Holland, Amsterdam, pp. 103–117.

King, J. L. and K. Lyytinen (2004) Reach and grasp, *MIS Quarterly* 28 (4), pp. 539–551.

Kuechler, W. and V. Viashnavi (2008) The emergence of design research in information systems in North America, *Journal of Design Research* 7 (1), pp. 1–16.

Layton, E. T. Jr. (1974) Technology as knowledge, *Technology and Culture* 15, pp. 31–41.

Lehtovuori, J. (1973) *Liiketaloustieteen metodologista taustaa*, Turun kauppakorkeakoulun julkaisuja, AI-6.

Lyytinen K. and J. L. King (2004) Nothing at the center? Academic legitimacy in the information systems field, *Journal of the AIS* 5 (6), pp. 220–246.

Lyytinen, K. and G. M. Rose (2003) The disruptive nature of information technology innovations: the case of Internet computing in systems development organizations, *MIS Quarterly* 27 (4), pp. 557–595.

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

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

McKay, J. and P. Marshall, P. (2007) Science, design, and design science: seeking clarity to move design science research forward in information systems, in *Proceedings of the Eighteenth Autsralasian Conference on Information Systems*, pp. 604–614.

McKay, J., P. Marshall, and G. Heath (2008) An exploration of the concept of design in information systems, in *Information Systems Foundations: Answering the Unanswered Questions About Design Research, The 4th ANU Workshop on Information Systems Foundations*, Australian National University, Canberra, Australia.

Niiniluoto, I. (1993) The aim and structure of applied research, *Erkenntnis* 38, pp. 1–21.

Niiniluoto, I. (1999) *Critical Scientific Realism*, Oxford University Press, Oxford.

Nunamaker, J. F., M. Chen, and T. D. M. Purdin (1990–1991) System development in information systems research, *Journal of Management Information Systems* 7 (3), pp. 99–106.

Orlikowski, W. J. and C. S. Iacono (2001) Research commentary: desperately seeking the "IT" in IT research – a call theorizing the IT artifact, *Information Systems Research* 12 (2), pp. 121–134.

Pitt, J. C. (2000) *Thinking about Technology, Foundations of the Philosophy of Technology*, Seven Bridges Press, New York, NY.

Popper, K. (1978) *Three Worlds*, The Tanner Lectures on Human Values, Delivered at the University of Michigan, Ann Arbor, MI.

Rogers, E. M. (1995) *Diffusion of Innovations*, 4th edn., The Free Press, New York.

Simon, H. (1969/1981/1996) *The Sciences of Artificial*, MIT Press, Cambridge, MA.

Strasser, S. (1985) *Understanding and Explanation Basic Ideas Concerning the Humanity of the Human Sciences*, Duquesne University Press, Pittsburg, PA.

Swanson, B. (1994) Information systems innovation among organizations, *Management Science* 40 (9), pp. 1069–1092.

van Aken, J. E. (2004) Management research based on the paradigm of design sciences: the quest for field-tested and grounded technological rules, *Journal of Management Studies* 41 (2), pp. 219–246.

van Aken, J. E. (2005) Valid knowledge for the professional design of of large and complex design processes, *Design Studies* 26**,** pp. 379–404.

Venkatesh, V., M. G. Morris, G. B. Davis, and F. D. Davis (2003) User acceptance of information technology: toward a unified view, *MIS Quarterly*, 27 (3), pp. 425–478.

Verschuren, P. and R. Hartog (2005) Evaluation in design-oriented research, *Quality & Quantity* 39, pp. 733–762.

von Hippel, E. (1988) *The Sources of Innovation*, Oxford University Press, Oxford.

von Hippel, E. (2005) *Democratizing Innovation*, The MIT Press, Cambridge, MA.

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

Winter, R. (2008) Design science research in Europe, *European Journal of Information Systems* 17, pp. 470–475.