8-2010

Design Science Research: Paradigm or Approach?

Sven Weber
Goethe University Frankfurt, svweber@wiwi.uni-frankfurt.de

Design Science Research: Paradigm or Approach?

Sven Weber
Goethe University Frankfurt
svweber@wiwi.uni-frankfurt.de

ABSTRACT

Due to the significant increase of theoretical evaluation of software prototypes, design science research (DSR) as a new research direction has emerged in recent years with the aim to ensure for both, rigor and relevance in prototyping research projects. On the one hand, a theoretical background ensures a usable and professional software prototype and on the other hand, new and innovative software prototypes provide rich data for theory testing and evaluating. DSR has proven to produce practically relevant research results but unfortunately it is still not a fully accepted research approach since it has somehow failed to develop theoretical contributions. Nevertheless, we believe that design science research is an important key factor for a new and innovative research paradigm. This paper provides an overview of DSR and tries to combine both, rigor and relevance, in a unified perception.

Keywords

Design Science Research, Research Paradigms, Literature Review.

INTRODUCTION

The challenge that IS researchers often face is to generate scientifically sound new knowledge while producing relevant research results that can be used by practitioners at the same time, e.g., in form of an IT artifact. DSR (March and Smith, 1995; Walls et al., 1992) has proven to produce practically relevant research results but is still not a fully accepted research approach since it has somehow failed to develop theoretical contributions (Hevner et al., 2004). Moreover, discussions about a DSR driven paradigm have occurred in the last decades to strengthen the theoretical underpinning of DSR (Gregg et al., 2001). DSR was derived from the engineering discipline and has therefore much older roots than the discussion itself. The emerging question is whether DSR depicts a separated paradigm or DSR is an approach that can be combined with other ones? In the light of this discussion several attempts have been made to combine both the original roots of DSR and the development of theoretical contributions. However, the methods literature in this area is still in its infancy. For instance, DSR has been combined in the past with action research (e.g., Allen et al., 2000) and with ethnography (e.g., Baskerville and Stage, 2001) in order to enhance the value and theoretical contribution of those projects. Other researchers combined DSR with behavioral science (e.g., Goldkuhl, 2004; Holmström et al., 2009) in order to create a theoretical contribution to the domain of study. Goldkuhl (2004) offers an approach how to use techniques of behavioral science in a DSR project. He presents three different types of grounding: internal, empirical, and theoretical, that can enhance a DSR project to generate grounded practical knowledge. Another study finds that both research strategies complement each other well (Holmström et al., 2009). In particular, they developed a framework how DSR as an exploratory research approach can be complemented by a second research cycle including the development of substantive and formal theory in order to make a contribution to the knowledge base besides focusing entirely on the problem solution and the IT artifact.

These examples show the potential of DSR to be combined with other research approaches. The research question of this paper is if DSR has the potential to become a new and independent paradigm or if it is an approach that fits into existing paradigms. Therefore we conducted a literature review about the most important DSR articles from recent years and analyzed the core characteristics of DSR.

The remainder of this paper is structured as follows: The next section outlines DSR as a research approach and thereby provides the basic characteristics of it. The following section describes the contrary position and outlines the potential of DSR to create an independent paradigm. Section 4 provides a literature overview about the most important DSR articles and their impacts on the ongoing discussion. Finally, the paper summarizes the insights and provides a recommendation for the DSR usage.

DESIGN SCIENCE RESEARCH AS A RESEARCH APPROACH

DSR has its roots in the architecture and engineering discipline. Hence, DSR attempts to create things that serve human purposes and thereby creates utility for the stakeholders (March and Smith, 1995). In contrast, natural and social science try to understand the reality and do not primarily focus on usefulness or direct applicability of their findings. Scientists using the DSR approach build IT artifacts to consider the relevance of the IT artifact for business requirements and thereby aim to define a problem solution (Au, 2001; Hevner et al., 2004). From this point of view, DSR can be seen as another research approach to solve practically relevant problems (McKay and Marshall, 2005).

IT artifacts are naturally occurring and always embedded in some place, time and community (Orlikowski and Iacono, 2001). As a result, they are highly dynamic within the environment. DSR concentrates on IT artifacts which are encompassing implementations, algorithms, mathematical equations etc. (Alter, 2008; Benbasat and Zmund, 2003; Hevner et al., 2004). The focus of an IT artifact lies on the problem itself. It is finished when it satisfies the requirements of all stakeholders and solves the relevant problem. On the one hand, it is necessary to understand why an IT artifact works or does not work while on the other hand it is necessary to understand how the IT artifact was created (Hevner et al., 2004).

To understand how an IT artifact was created we have to understand the underlying kernel theories of the IT artifact. Those kernel theories are defined by the evaluation and modification of the natural and social science theories. Moreover, they are created by the experience and creativity of the researcher (Hevner et al., 2004; Markus et al., 2002; Walls et al., 1992). Therefore, DSR is both: developing new solutions of IT artifacts to existing but unsolved problems and/or matching solutions to new and unsolved problems (Holmström et al., 2009).

Figure 1 depicts a possible DSR approach which is a combined framework drawn from existing approaches in DSR literature (Table 1). According to March and Smith (1995) as well as Walls et al. (1992) DSR encompasses processes and products which are both derived from kernel theories (Hevner et al., 2004; Markus et al., 2002; Walls et al., 1992). These kernel theories are influenced and influencing the requirements of the IT artifact and therewith the problem solution. The connection between the evaluation and instantiation ensures a high utility of the IT artifact while the requirements are satisfied at the same time.

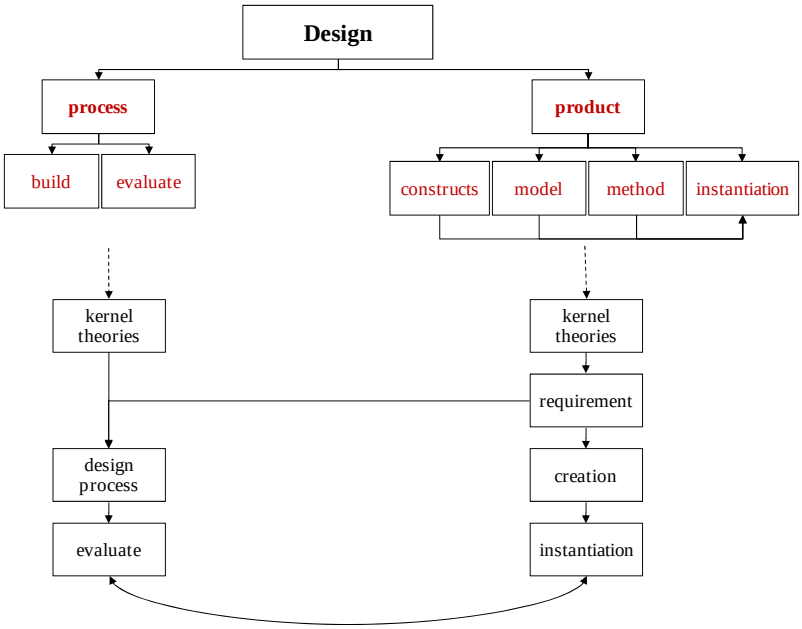

Figure 1. DSR framework according to Hevner et al. (2004), March and Smith (1995) as well as Walls et al. (1992)

Processes are distinguished into two basic elements: the building and evaluation of the IT artifact (Baskerville et al., 2009; Hevner and March, 2003; March and Smith, 1995). Building an IT artifact is a sequence of activities to produce 'something new'; an innovative product that solves a real world problem. The evaluation of an IT artifact provides feedback to the design process and generates new knowledge about the problem at hand. The newly generated insights serve to improve both the quality of the IT artifact and the design process itself (Hevner et al., 2004). The build and evaluate elements are conducted partly in parallel and involve multiple iterations. Through these multiple iterations, the IT artifact is fully generated to the

satisfaction of the researchers and practitioners that later make use of it (Markus et al., 2002). Thereby, DSR creates utility and a meaningful contribution to practice (Hevner et al., 2004; March and Smith, 1995; McKay and Marshall, 2005; Walls et al., 1992). However, as Hevner et al. (2004) criticize, DSR frequently fails to make a scholarly contribution to the knowledge base in the domain of study.

According to March & Smith (1995) the relevant end products (i.e. IT artifacts) are either constructs, models, methods, instantiations, or a combination thereof. This end product is developed from the design cycle (Takeda et al., 1990). Figure 2 depicts constructs as basic elements of instantiations. These constructs can be seen as the vocabulary of a domain. They constitute a conceptualization to describe problems within the domain to specify their solution. Analyzing relationships between constructs form the basis for the construction of models. Models are sets of propositions or statements that depict these relationships between constructs. By the construction of a model, representing a kind of automata theory (Hopcroft and Ullman, 1979), a representation of the real-world is created. Moreover, with such a real-world representation, new constructs can be created or old ones can be improved. A method is a set of steps (algorithm or guideline) to solve a defined problem. More precisely, a method is a formal implementation of constructs as well as models and at the same time a representation of the solution space. An instantiation is the realization of an IT artifact in its environment, e.g. a prototype.

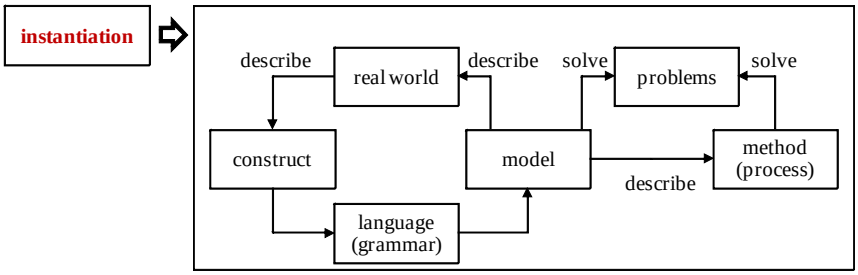

Figure 2. IT artifact instantiation.

Due to the presented framework, new IT artifacts can be created from kernel theories. Thereby these IT artifacts are theorized to contribute to the existing knowledge base (Carroll and Kellogg, 1989; Orlikowski and Iacono, 2001). In addition, these IT artifacts can be improved by several evaluation steps until the postulated requirements are reached. Summing up so far, the DSR framework can be regarded as an approach to create and evaluate an IT artifact that provides a real world problem solution. The depicted procedure of constant improvement can be seen on a higher level of abstraction as we will illustrate in the next section where we will discuss research paradigms.

DESIGN SCIENCE RESEARCH AS A RESEARCH PARADIGM

According to Filstead (1979), a research paradigm is defined by a 'set of interrelated assumptions about the social world which provide a philosophical and conceptual framework for the organized study of that world'. In the context of this definition, the following section provides a brief overview over the most important research paradigms according to Gregg et al. (2001), Orlikowski and Baroudi (1991) as well as Robey (1996).

IS Research Paradigms

The supporters of the interpretive paradigm see the world as a social process (Orlikowski and Baroudi, 1991). Social systems cannot be regarded independently and without the influence of their members. Individuals, organizations or groups construct the social systems because every user has special requirements that must be addressed by the system. These researchers seek to understand and interpret the social process. The methodologies used in this paradigm are mainly qualitative research and the exposure of special sets of constructs to social effects (Gregg et al., 2001; Orlikowski and Baroudi, 1991).

The positivistic paradigm is the leading philosophical strand in today's IS research (Orlikowski and Baroudi, 1991). This paradigm is primarily based on an empirical world-view seeking to obtain knowledge through quantitative methods. The reason for this procedure is the notion that reality is existent only in one reality (Gregg et al., 2001). This reality is not disrupted by the action of humans, and social worlds are not influenced by the actions of their members. It consists of a social world that is controlled by nature and therefore can be regarded analogous to the natural world. The research efforts are independent from the analyzed object. Hence, the researcher plays only a passive role (Orlikowski and Baroudi, 1991).

These two research paradigms are able to embrace many research concepts of the IS research discipline. However, in our opinion they are not able to cover DSR completely.

Gregg et al. (2001) recommended to introduce a third paradigm that exists in harmony with the positivistic and interpretive paradigm. This paradigm is called the socio-technologist or developmentalist paradigm and is able to give an answer to the missing theoretical contribution of DSR. In most other cases the technology and software development is seen as a value which is present or not. In this paradigm, a high responsibility lies on the construction and evaluation phase of technology and software. Hence, when this software is missing some important issues within organizations cannot be solved. The socio-technologist or developmentalist paradigm focuses on the creation of technology and the technology itself to affect individual and organizational experience in a positive manner. Moreover, the IS environment can be seen as a social implemented system. According to Gregg et al. (2001), this paradigm is equivalent with DSR because of its developmental nature.

Similarities and Differences between the Three IS Research Paradigms

IS research is quite an academic discipline and uses a theoretical founded base to improve and promote knowledge (Gregg et al., 2001). In contrast, IS development by itself is quite a practical discipline and does not necessarily need research (Gregg et al., 2001; Mantei and Teorey, 1989; Nunamaker et al., 1991). However, a wide stream of literature has assumed those definitions and explored that IS research and IS development have lots of similarities (Gregg et al., 2001; Holmström et al., 2009). According to Nunmaker et al. (1991) research is represented by its objectives and methods, whereby the objectives require a multi methodological approach to integrate theory building, system development, and experimentation. In contrast, IS development is defined as 'the art of building software' but also as the corner stone for research methodology (Nunamaker et al., 1991). Subsequently, the research methodologies can be subdivided and classified as part of a certain research paradigm (Figure 3).

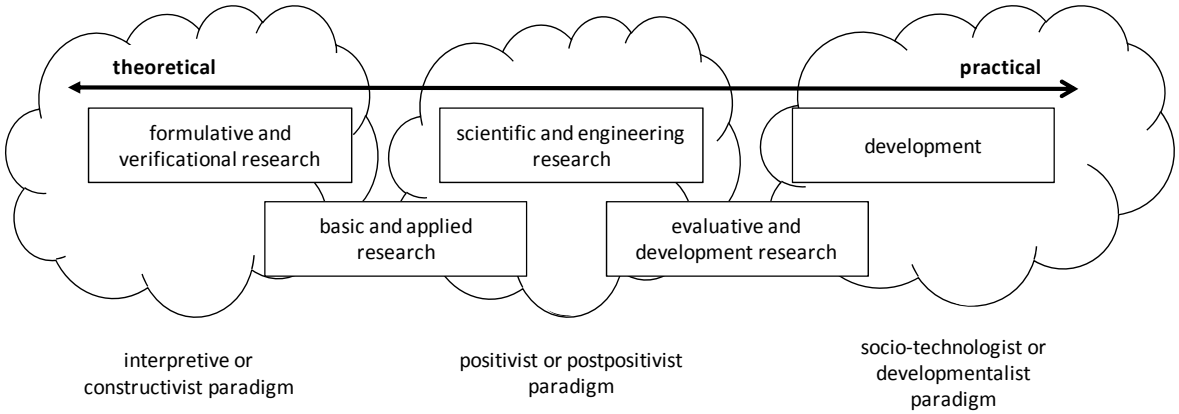

Figure 3. Perceptions of research objectives and methods.

Figure 3 is derived from Nunmaker et al. (1991). The different perceptions are organized on a theoretical scale. The scale illustrates the degree of theoretical importance on the one side versus practical relevance on the other side. The most theoretical perception is the 'formulative and verificational research' according to Nunmaker et al. (1991). Its goal is to gain insights and improve the understanding of the problem area. Subsequently, the 'basic and applied research' develops and tests for theoretical goals/reasons, theories and hypotheses. These two perceptions are combined in the interpretive or constructivist paradigm (Gregg et al., 2001).

Starting from the other side the 'development' perception represents the most practical methodology. According to Nunmaker et al. (1991), it provides 'the systematic use of scientific knowledge' to build, evaluate and develop new technologies or prototypes. Whereby, it is not only determined to the creation process but also it 'improves the effectiveness and efficiency of processes at the individual and organizational level' (Gregg et al., 2001). The 'evaluative and development research' is located closely to the 'development' and includes both: the evaluative (more theoretical) and the developmental (more practical) approach. Both form the socio-technologist or developmentalist paradigm (Gregg et al., 2001).

At last the 'scientific and engineering research' represents the connection between the theoretical and practical approaches. This perception and the linking perceptions 'basic and applied research' and 'evaluative and development research' are represented by the positivist or postpositivist paradigm.

The example of Nunmaker et al. (1991) and especially Figure 3 outlines that the IS research paradigms cannot be completely separated by their methodologies and approaches. On the one hand, the major part of these methodologies and approaches encompass an interdisciplinary process that fits to more than one paradigm. On the other hand, each paradigm inhibits a key

approach which depicts its core research direction. For instance, basic and applied research with a data collection related to interviews and a strict focus on theory-building goals can be clearly sorted into the interpretive paradigm.

In the second section we discussed that DSR is derived from the engineering discipline and depicts the science of the artificial (March and Smith, 1995; Simon, 1969). Thereby, the major goals of DSR are the building and evaluation of an IT artifact to solve a real world problem (Hevner et al., 2004). As a consequence, DSR can be sorted into the socio-technologist or developmentalist paradigm (Gregg et al., 2001). In our opinion, this conclusion is true but DSR is not equal to this paradigm. DSR is not matured enough to represent the whole developmentalist paradigm but has the potential to support it. We conclude that DSR is one possible approach of the socio-technologist or developmentalist paradigm but is not limited to it. DSR is most effective when the researcher and practitioners shift between pragmatic and critical realist perspectives (Bunge, 1984). Thereby, an important step of DSR is the evaluation of the IT artifact. Hence, DSR can also encompass elements of a positivist paradigm (e.g. for the evaluation) or an interpretive paradigm (e.g. for the identification of the initial requirements). These directions depend on the researcher's point of view and his aspired goals.

DESIGN SCIENCE RESEARCH LITERATURE OVERVIEW

This section provides a brief overview of the cited articles in this paper. As mentioned before, DSR has its roots in the architecture and engineering discipline. Hence, the idea of DSR is much older than the literature has named it. The first literature based definition of design was provided by Simon (1969). Approximately 25 years later, the term DSR was firstly mentioned by Walls et al. (1992). 12 years later Hevner et al. (2004) wrote a refined definition of DSR and illustrated the DSR approach based on 7 guidelines. Table 1 depicts an overview of the most important DSR literature of the last 40 years. The provided literature overview maps directly to the mentioned paradigm discussion beforehand and confirms the intention that DSR is a research approach rather than a research paradigm.

Reference	Topic	Description
Simon (1969)	Design	Provides a discussion on the design of the artificial.
Mantei and Teorey (1989)	IS Development	Describes the IS developmental life cycle on the example of a user interface for a DBMS.
Walls et al. (1992)	DSR basic literature	Depicts a milestone of DSR and provides a first DSR framework. Moreover, this article contains basic definitions of DSR and of IT artifacts
March and Smith (1995)	DSR basic literature	Provides basic definitions of DSR and a framework for creating an IT artifact It is the first article that tries to place DSR research in context to other disciplines and paradigms.
Orlikowski and Iacono (2001)	DSR basic literature	A detailed description of an IT artifact.
Hevner et al. (2004)	DSR basic literature	Depicts a milestone of DSR. Basic definitions of DSR and IT artifacts are provided in this article. Moreover, it conceptualizes a new framework for creating an IT artifact (7 Guidelines for DSR in Information Systems Research). Additionally, it evaluates the guidelines on the basis of three different Design Research (DR) papers.
Carlsson (2006)	DSR general literature	Provides a definition of DSR derived from Hevner et al. (2004) and Walls et al. (1992). Additionally, it classifies DSR as critical realism.
Baskerville (2008)	DSR general literature	A short introduction of DSR and IT artifacts.
Winter (2008)	DSR general literature	A short introduction of Design Science (DS) and Design Research (DR).

Kuechler and Vaishnavi (2008)	DSR general literature	Describes kernel theories and suggests a framework for DSR.
March and Storey (2008)	DSR general literature	Provides a definition of DSR and 5 examples of literature that uses the DSR approach.
Markus et al. (2002)	DR	Applies the DSR approach to the example of TOP Modeler (an IT artifact that supports emergent knowledge processes).
Aalst and Kumar (2003)	DR	Describes the creation of a new IT artifact (XML-based schema for inter-organizational workflow).
Weedman (2008)	DR	Describes the creation of a new IT artifact (Sequoia 2000; the IT artifact brought together prestigious scientists from computer science and earth science to get in collaboration).
Umapathy et al. (2008)	DR	Describes the creation of a new IT artifact (embedding a mechanism that helps designers develop integration solutions based on recurring solutions captured as patterns).
Gregg et al. (2001)	DSR vs. behavioral science	Classifies DSR as a separated paradigm called socio-technologist or developmentalist paradigm.
Holmström et al. (2009)	DSR vs. behavioral science	Describes the difference of DSR and behavioral science. Thereby, it explores 4 phases which are derived from the different approaches.

Table 1. Important DSR literature

CONCLUSION

In this paper we analyzed DSR in terms of its methodology. On the one hand, we found that DSR serves the demand about rigorous and relevant research projects and on the other hand that DSR is still not a fully accepted research approach or independent paradigm. DSR is derived from the engineering discipline and depicts the science of the artificial (March and Smith, 1995; Simon, 1969). Additionally, IT artifacts are always embedded in some place, time, and community (Orlikowski and Iacono, 2001) and therefore not an overall description of the world. Moreover, the major goals of DSR are to build and evaluate those IT artifact to solve a real world problem (Hevner et al., 2004). Building an IT artifact reflects directly to the socio-technologist or developmentalist paradigm but not necessarily the following evaluation part. This part can be conducted in a technical and therewith developmental, in a quantitative and therewith positivistic, or in a qualitative and therewith interpretive manner. This example shows the full potential of a DSR approach to combine the advantages of different paradigms. For instance, according to McKay and Marshall (2005), DSR has 'not to put […] in a positivistic box in which all elements of interest are perfectly and immutably defined […]'. DSR can also be conducted in a developmental manner by focusing on the creation part, or in an interpretive manner by generating theory out of the created IT artifact and its usage. Especially the combination with interpretive methods is still not fully explored but we see high potential for a combined use of behavioral science and DSR.

Our conclusion is that DSR can be used as an approach in many different paradigms such as positivist, interpretive, or developmentalist apart from its engineering roots. Thus, DSR outlines his potential to be combined with other approaches in different research paradigms, e.g. ethnography (Baskerville and Stage, 2001) or behavioral science (Goldkuhl, 2004; Holmström et al., 2009). Overall, by naming one paradigm to describe the DSR approach, one can say that DSR can be sorted into the socio-technologist or developmentalist paradigm (Gregg et al., 2001). Unfortunately, Gregg et al. (2001) explored not the potential of DSR to be combined with other methods and judged DSR to its developmental nature. Thereby, they defined DSR equally to the socio-technologist or developmentalist paradigm. As mentioned before, we see more potential in the DSR approach and the research question if DSR depicts an approach or paradigm is answered: DSR is a pluralistic research approach that cannot and should not be separated in an existing research paradigm. The articles depicted in Table. 1

supports our argumentation and additionally supports the hypothesis that DSR is on its way to a fully accepted and pluralistic research approach. Additionally, by using a pluralistic research approach, DSR can ensure a theoretical contribution to the domain of study.

We hope to contribute to the existing discussion about DSR with this article. Moreover, we hope to provide a recommendation for researchers who are potentially willed to use DSR approaches in their projects. From a scientifically point of view, the DSR approach is still in its infancies but has to be strengthened by more pluralistic research projects.

REFERENCES

1. Aalst, W. and Kumar, A. (2003) XML-Based Schema Definition for Support of Interorganizational Workflow, *Information Systems Research*, 14, 1, 23-46.
2. Allen, D.K., Colligan, D., Finnie, A. and Kern, T. (2000) Trust, Power and Interorganizational Information Systems: The Case of the Electronic Trading Community TransLease, *Information Systems Journal*, 10, 1, 21-40.
3. Alter, S. (2008) Defining Information Systems as Work Systems: Implications for the IS field, *European Journal of Information Systems*, 17, 448-469.
4. Au, Y.A. (2001) Design Science I: The Role of Design Science in Electronic Commerce Research, *Communications of the Association for Information Systems*, 7.
5. Baskerville, R. (2008) What Design Science is Not, *European Journal of Information Systems*, 17, 5, 441-443.
6. Baskerville, R., Pries-Heje, J. and Venable, J. (2009) Soft Design Science Methodology, *Proceedings of the Design Science Research in Information Systems and Technology DESRIST*, Philadelphia. Pa.
7. Baskerville, R. and Stage, J. (2001) Accommodating Emergent Work Practices: Ethnographic Choice of Method Fragments, Kluwer Academic Publishers, New York.
8. Benbasat, I. and Zmund, R.W. (2003) The Identity Crisis within the IS Discipline: Defining and Communicating the Discipline's Core Properties, *MIS Quarterly*, 27, 2, 183-194.
9. Bunge, M., Philosophical Inputs and Outputs of Technology, in: G. Bugliarello, D. Donne (Eds.) History and Philosophy of Technology, University of Illinois Press, 1984, pp. 263-281.
10. Carlsson, S.A. (2006) Towards an Information Systems Design Research Framework: A Critical Realist Perspective, *Proceedings of the Proceedings of the 1st International Conference on Design Science Research in Information Systems and Technology*, Claremont, CA.
11. Carroll, J. and Kellogg, W. (1989) Artifact as Theory Nexus: Hermeneutics Meets Theory-Based Design, *Proceedings of the CHI '89*.
12. Filstead, W.J. (1979) Qualitative Methods: a Needed Perspective in Evaluation Research, Sage, Beverley Hills.
13. Goldkuhl, G. (2004) Design Theories in Information Systems - A Need for Multi-Grounding, *Journal of Information Technology Theory and Application*, 6, 2, 59-72.
14. Gregg, D.G., Kulkarni, U.R. and Vinze, A.S. (2001) Understanding the Philosophical Underpinning of Software Engineering Research in Information Systems, *Information Systems Frontiers*, 3, 2, 169-183.
15. Hevner, A., March, S., Park, J. and Ram, S. (2004) Design Science in Information Systems Research, *MIS Quarterly*, 28, 1, 75-105.
16. Hevner, A.R. and March, S.T. (2003) The Information Systems Research Cycle, *IT Systems Perspective*, 36, 11, 111-113.
17. Holmström, J.B., Ketokivi, M. and Hameri, A.P. (2009) Bridging Practice and Theory: A Design Science Approach, *Decision Science*, 40, 1, 65-87.
18. Hopcroft, J.F. and Ullman, J.D. (1979) Introduction to Automata Theory, Languages and Computation, Addison Wesley Reading.
19. Kuechler, B. and Vaishnavi, V. (2008) On Theory Development in Design Science Research: Anatomy of a Research Project, *European Journal of Information Systems*, 17, 5, 489-504.
20. Mantei, M. and Teorey, T. (1989) Incorporating Behavioral Techniques into the System Development Life Cycle, *MIS Quarterly*, 13, 3, 257-274.
21. March, S.T. and Smith, G. (1995) Design and Natural Science Research on Information Technology, *Decision Support Systems*, 15, 4, 251-266.
22. March, S.T. and Storey, V.C. (2008) Design Science in the Information Systems Discipline: An Introduction to the Special Issue on Design Science Research, *MIS Quarterly*, 32, 4, 725-730.
23. Markus, M.L., Majchrzak, A. and Gasser, L. (2002) A Design Theory for Systems that Support Emergent Knowledge Processes, *MIS Quarterly*, 26, 3, 179-212.
24. McKay, J. and Marshall, P. (2005) A Review of Design Science in Information Systems, *Proceedings of the 16th Australasian Conference on Information Systems*, Sydney, Australia.

25. Nunamaker, J.F., Chen, M. and Purdin, T.D.M. (1991) Systems Development in Information Systems Research, *Journal of Management Information Systems*, 7, 3, 89-106.
26. Orlikowski, W. and Baroudi, J. (1991) Studying Information Technology in Organizations Research Approaches and Assumptions, *Information Systems Research*, 2, 1, 1-28.
27. Orlikowski, W.J. and 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, 121-134.
28. Robey, D. (1996) Research Commentary: Diversity in Information Systems research: Threat, Promise, and Responisbility, *Informatiokn Systems research*, 7, 4, 400-408.
29. Simon, H.A. (1969) The Science of the Artificial, MIT Press, Cambridge, MA.
30. Takeda, H., Veerkamp, P., Tomiyama, T. and Yoshikawam, H. (1990) Modeling Design Processes, *AI Magazine*, 37-48.
31. Umapathy, K., Purao, S. and Barton, R.R. (2008) Designing Enterprise Integration Solutions: Effectively, *European Journal of Information Systems*, 17, 5, 518-527.
32. Walls, J.G., Widmeyer, G.R. and El Sawy, O.A. (1992) Building an Information System Design Theory for Vigilant EIS, *Information Systems Research*, 3, 1, 36-59.
33. Weedman, J. (2008) Client as Designer in Collaborative Design Science Research Projects: What does Social Science Design Theory tell us?, *European Journal of Information Systems*, 17, 5, 476-488.
34. Winter, R. (2008) Design Science Research in Europe, *European Journal of Information Systems*, 17, 5, 470-475.