

Danish University Colleges

Scientific Methods 2 Investigations in Software Development

Nordbjerg, Finn Ebertsen

Publication date:
2021

[Link to publication](#)

Citation for pulished version (APA):
Nordbjerg, F. E. (2021, Feb 5). Scientific Methods 2: Investigations in Software Development.

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

Download policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Scientific Methods 2

Investigations in Software Development

Finn E. Nordbjerg

University College of Northern Denmark, 2021

fen@ucn.dk

Abstract

This paper briefly presents an approach to apply Design Science Research in Software Development. The approach is based on combining theory and practice through reflection. The paper is part of a series of lectures notes that are written to support the course “Investigation and Professional Communication” at the Professional Bachelor Programme in Software Development at UCN.

Acknowledgement: Thanks, to my colleague Esben Skov Laursen, UCN Design and Production for valuable discussions and advises.

1 Introduction

As you may recall from [1], software development takes place in a space described by the ITU Triangle [2]. Figure 1 illustrates this:

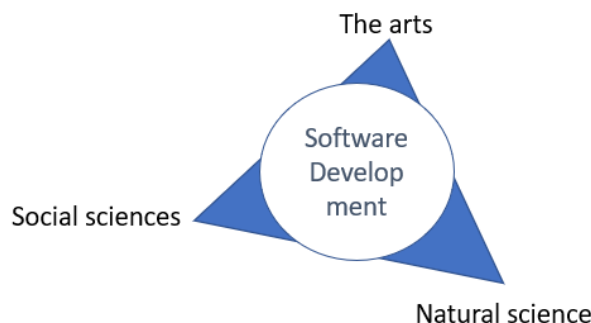


Figure 1: The ITU-Triangle [2]

When we are gathering requirements for a software system, we are working in the field of the arts and business. When we are designing, implementing, and testing we are in the field of natural science (usability testing may use methods from the arts and social science). Software is a man-made artifact, but we are using methods from natural science. In addition, we want to investigate the creation of software. Also, much research into software development is carried out in close connection with practice, theory and practice are going hand-in-hand.

The study of the creation process may be carried out by using what is known as *Design Science Research* – DSR (see figure 2):

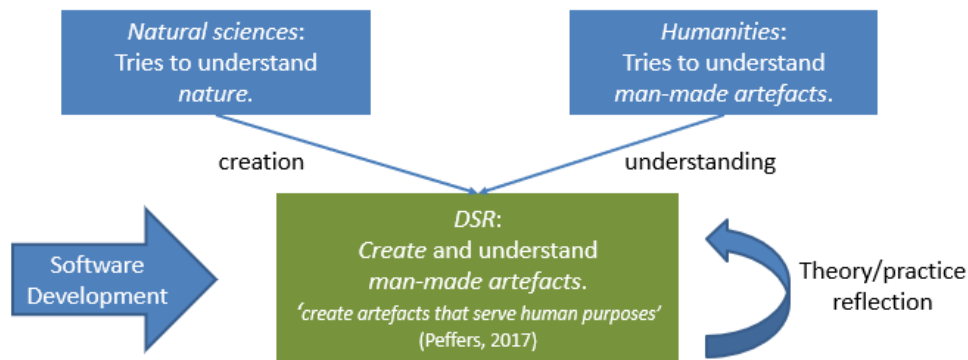


Figure 2: Design Science Research (based on [3]).

In DSR the creation (development) and understanding are combined. Normally, there are two parallel streams of work: *practice stream* and *knowledge stream*. The two streams are connected through reflection (figure 3).

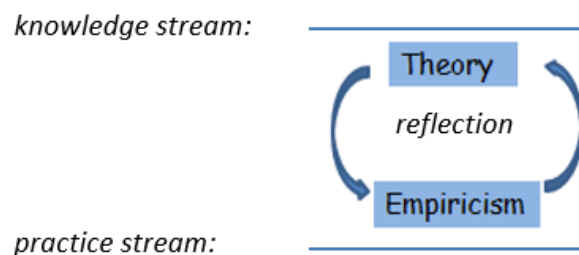


Figure 3: Interaction between practice and theory.

The idea is to generalise the practical development problem into a more general research question. Although the research question should be general, it is important to realise that it should not be so general that it has no practical value. Hence, when generalising it is important to keep the question within the context of the practical problem. So the research question may only be general within a given practical context. In the next section we will take a closer look at DSR.

At UCN, we have a research group that works with Big Data Analysis. We design our research projects in collaboration with industry and try to combine the practice stream and knowledge stream. Figure 4 illustrates the design:

In the recent period, we have been working on a project, where the development project aims to build a system that collects sensor data from a manufacturing plant and analyses the data in order to assist the management of the manufacturing process [4]. The development includes several areas of knowledge (for instance: Big Data, Business Intelligence, System architecture and visualisation of production data). We have focussed on the knowledge stream within big data, but

we have begun focussing on scalable architecture and plan to add a new knowledge stream about visualisation, where it should be looked into how AR¹ and VR² technologies can be used to display

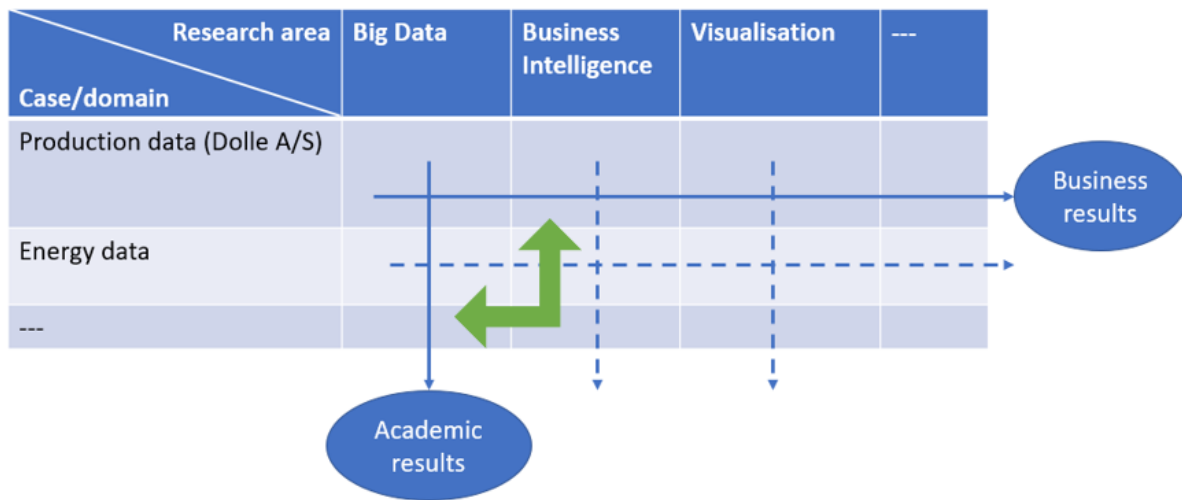


Figure 4: Example of research design.

results of data analysis to different user groups (machine operators, production managers and planners). It is very important that the two streams are closely connected, walk hand-in-hand and are mutually enriching each other. The practice stream is developing a useful software package for the business we are cooperation with. Previously, we have been working on energy data. where we were able to achieve good academic results (see for instance [5]), but we could not find a committed business partner, so we shifted to production data. This underlines the importance of having a practice stream as well. Now, we are almost done with the development project (practice stream) and the next phase of the research will be to generalise the developed solution to a framework that can be used to build solutions for other manufacturing companies, and then test that framework on other companies. In this work, it is important not to generalise too much; the framework must be able to give practical value to other enterprises in the same sector. So, the generalisation must be done within the context of the practice. In addition, we are going to generalise our experience in cooperation with the company to recommendations for general transition process for other companies that want to begin to use their production data. The latter is an example of a “meta knowledge stream” or a “process stream” that often may be added when working with Small and Medium sized Enterprises (SMEs): It is often the first time that a SME is participating in a R&D project, so for both the R&D group and the employees of the SME it is a learning process – how can we improve our cooperation with the SME, and what can we learn in order to improve the process for other SMEs? In our case, the SME is embarking on a voyage into Industry 4.0 and utilising production data – a completely new world for many SMEs in the manufacturing sector.

Finally, it is worth noting that in study (and some research) projects, one may lack the industrial partner. In that case the practice stream will be developing a prototype, but it is preferable to have a real industrial partner with real problems.

¹ Augmented reality

² Virtual reality

2 Design Science Research – basic principles

The thinking behind Design science research (DSR) is not new. The first presentation that I have come around is [6]. DSR is proposed as a methodology for research into software development in [3]. A shorter and more readable presentation can be found in [7]. The following is based on [7].

Firstly, it is important to distinguish between (product) development and research. One may develop a commercially very successful product, but it does not give any new knowledge, and hence, does not meet the criteria for research. Research is characterised by addressing a general acknowledged problem, building upon existing knowledge (published in literature), and making a new and original contribution to knowledge within the field [7]. In the following, we shall look at these criteria:

2.1 Problem driven

Not all practical development problems are suitable for research. The problem must be complex and of general nature, so that the insight achieved through studying the problem may contribute to solve other practical problems of a similar nature. So, the problem must be generalisable. In product development many problems are simpler and more specific to the concrete development task. But recall from the previous sections that one must be careful when generalising. If the problem is simplified too much in the process of generalising, then the general solution may become with no or little practical value. Using an analogy from math, do not try to generalise by looking for the intersection of different concrete instances of the problem. One should rather look for a generalisation that includes the union of these instances. See figure 5.

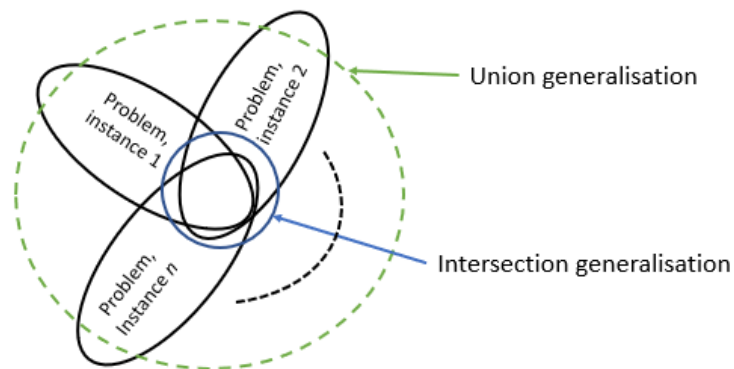


Figure 5: Generalisation: Go for Union intersection that embraces all thinkable instances of the problem, not intersection that will be too general, too simplified and have too little practical value.

2.2 Building on existing knowledge

All research builds upon existing knowledge and previous research. Research is published in journal or conference papers or in monographies – id. in literature. Hence, it is important to place the problem in context: what is already known and published about it? Why is this an important problem to investigate? What contributions to the total body of knowledge about software development will the investigation give?

So, the research must include literature review and take outset in existing knowledge.

2.3 Contribute with new knowledge

If an inquiry is to be research, and not just part of a development process, then it must be clear how the results of the inquiry contribute to the general knowledge in the field. These contributions may be in many different ways, e.g., applying a known theory or methodology to a new problem domain, combining known solutions in a novel way to solve a new kind of problems or generalising a solution, so it may be applied to other similar problems.

As a concrete example of this is: In [8] an investigation conducted using the DSR thinking is reported. The problem domain is data from field spraying in farming. The data are collected on small devices on the tractors spraying the field. Normally, these data will be transmitted in raw form to a server. Cleansing, validation, and aggregation will be done on the server. In [8] we propose a solution, where part of this processing is done on the device, using the device's capability of running SQLite and doing the processing in standard SQL and hereby reducing network traffic and server load. So well-known technology (SQLite) is used in the process of cleansing and aggregating data in a big data problem and hereby reducing network traffic and server load. Some time later, this author is supervising a bachelor project, where the student was asked to look into, if and how costs for cloud hosting a big data solution could be reduced. This problem also included devices collecting data, in this case data on indoor climate. The raw data was transmitted to the cloud and all storing and processing was done there. The students was directed to look at [8] and with outset in this, the student developed a solution which reduced the hosting costs significantly. If the originally work has been a development project only, and not reported and published, then the solution would not have been available in the new context.

3 Design Science Research – guidelines

In literature there can be found many guidelines to DSR. [7] includes a comprehensive review. From that work, a number of central activities can be identified. Some of the activities are normal activities for a development project (practice stream), and others are part of the research stream. Both sets of activities are important and necessary and are supporting each other.

Figure 6 illustrates the workflow:

One may start with a practical problem in a development project or with a theory-based hypothesis. In all cases, the first step is to identify the problem (requirements or problem statement in the practice stream). It is important to remember that the problem must have practical relevance, i.e., its solution will add business value. So, the problem will normally be of a complex and interdisciplinary nature. Next phase is a desk research investigating what is already known about this problem and reported in literature? Are there practical solutions around? Is it a general problem? If it is decided that the problem is worth researching, then the problem statement is generalised into a research question (research stream).

Now, when the problem is clear and it is worth researching into, the development of the solution begins. Here one works on the practical and the conceptual solution hand-in-hand. The one is not more important than the other but do remember to reflect on and generalise your practical solutions. Often, one sets out by building a conceptual framework for the solution (research stream), analyse alternative solutions and design an architecture for the solutions. From this a prototype of the concrete solution may be created. Experiences from building the prototype will

give feedback to the conceptual solution, so this is an iterative and agile process shifting between practice and theory (figure 3).

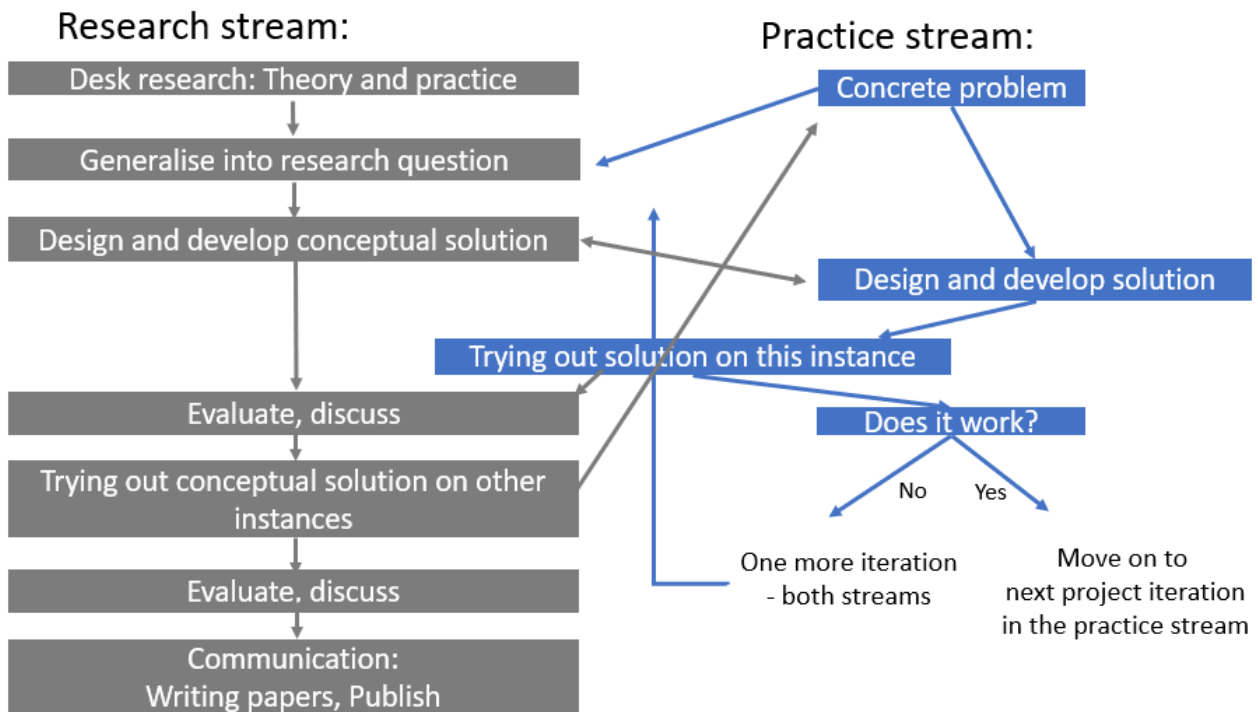


Figure 6: DSR workflow.

Then the concrete solution is to be tested on the actual instance of the problem, just like one normally would test a software solution. If the solution passes the tests on this instance of the problem, then one is ready for next iteration in the practice stream. Otherwise, one must go back in both streams.

In the research stream, testing is not enough, here the solution must be evaluated theoretically, is it supported by general recognised theory, by other studies etc.?

At this stage one may want to publish one's preliminary results in an article or a conference paper.

But to gain more confidence in the solution, the solution should be tried on other instances of the problem.

Hereafter, evaluation and discussion remains. Did the solution work well? How general is it? What are the limitations? What should be investigated further?

In addition, it should be added that if the practice stream is conducted in cooperation with real company and real users, then it is also important to consider including the meta knowledge stream mentioned in section 2, where the development process is considered. Experiences from the cooperation with the enterprise should be collected and reflected upon and generalised in order to learn and improve the working process. This process improvement should be an integrated part of any software development process [6].

4 Reflection

A central part of the DSR work cycle is reflection. It is through reflection that theory and practice are connected. So, let us take a closer look on what reflection is. According to [9] reflection includes elements as:

- *Making sense of experience.* When ever you have an experience, you will normally think about it trying to learn from it.
- Reflection requires you to “*stand back*”. It is not easy to reflect in the middle of an action.
- *Different points of view.* One goes over an experience several times looking at it from different points of view. One will try to balance the different points of view and take everything into account.
- *Clarity and understanding.* Through reflection you try to achieve a deeper understanding and insight into theories and concepts.

So, whenever observing something in your practice development project, step back and think about it. How do these observations align with the theory? And do the observations suggest modifications of the theory?

This thinking about improving your practice goes way back (Schön, 1983 [10]). In software development, the ideas are unfolded by the Danish computer scientist Lars Mathiassen [6]. In many modern software development methods this process of reflection is built in (for instance “retrospectives” in Scrum [11]).

5 Conclusion

In this paper, we have presented an approach to do investigations into software development. The approach tries to join solving a practical problem with the creation of new general knowledge. There are two (sometimes three) work streams: a *practice stream* and a *knowledge stream*. The practice stream leads the way, but the two streams are closely connected and each other’s preconditions, and they will mutually enrich each other. The connection between the streams is based on *reflection*. In [12] the work flow is summarised with this figure 7:

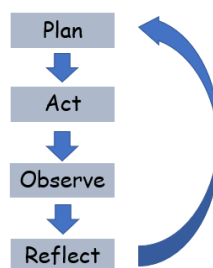


Figure 7: The DSR workflow [12]

In the practice stream, you plan and carry out an attempted solution to a problem (an experiment), you observe the outcome of the experiment. Through reflection you either modify the experiment or generalise the solution (shifting to the knowledge stream).

In addition, if the practice stream is executed in co-operation with an enterprise (in many cases a SME); it may be worthwhile to add the third stream: the *meta knowledge stream*, where the

experiences from the work process with the enterprise is considered, documented and generalised.

It is also important to note that the practical problem under investigation is not purely of a technical nature. In order to have practical value, it will be interdisciplinary. This is illustrated by the ITU Triangle (figure 1, [2]).

6 References

- [1] F. E. Nordbjerg, "Scientific Methods 1 Introduction," Aalborg, 2020.
- [2] M. Tofte, "The story of ITU," 2019. [Online]. Available: <https://en.itu.dk/about-itu/the-story-of-itu>. [Accessed: 19-Mar-2020].
- [3] K. Peffers, T. Tuunanen, M. A. Rothenberger, and S. Chatterjee, "A Design Science Research Methodology for Information Systems Research," *Publ. J. Manag. Inf. Syst.*, vol. 24, no. 3, pp. 45–78, 2007.
- [4] N. Iftikhar, T. Baattrup-Andersen, F. E. Nordbjerg, E. Bobolea, and P.-B. Radu, "Data Analytics for Smart Manufacturing: A Case Study," in *Proceedings of the 8th International Conference on Data Science, Technology and Applications (DATA 2019)*, 2019, p. pages 392-399.
- [5] N. Iftikhar, X. Liu, F. E. Nordbjerg, and S. Danalachi, "A Prediction-Based Smart Meter Data Generator," in *2016 19th International Conference on Network-Based Information Systems (NBIS)*, 2016, pp. 173–180.
- [6] L. Mathiassen, "Reflective Systems Development," *Scand. J. Inf. Syst.*, vol. 10, no. 1 & 2, pp. 67–118, 1998.
- [7] T. J. Ellis, Y. Levy, and F. Lauderdale, "A Guide for Novice Researchers : Design and Development Research Methods What Design and Development Research Is –," *Proc. Informing Sci. IT Educ. Conf.*, pp. 107–118, 2010.
- [8] N. Iftikhar, X. Liu, and F. E. Nordbjerg, "Relational-Based Sensor Data Cleansing," 2015, pp. 108–118.
- [9] S. Cottrell, *Skills for Success: Personal Development and Employability - Stella Cottrell - Google Bøger*, 3rd ed. Palgrave, 2015.
- [10] D. Schön, *The Reflective Practitioner How Professionals Think in Action*. Basic Books, 1983.
- [11] "What is a Sprint Retrospective?," *Scrum.org*, 2021. [Online]. Available: <https://www.scrum.org/resources/what-is-a-sprint-retrospective>. [Accessed: 18-Jan-2021].
- [12] J. Dudovskiy, "Action Research - Research Methodology," 2017. [Online]. Available: <https://research-methodology.net/research-methods/action-research/>. [Accessed: 16-Jan-2018].