Pragmatic Research Tactics for Starting IT Engineers

D. Greve[1]
Teacher, Researcher
University of Applied Sciences Utrecht
Utrecht, The Netherlands
E-mail: dan.greve@hu.nl

K. van Turnhout
Coordinator, Researcher
HAN University of Applied Sciences
Arnhem, The Netherlands
E-mail: koen.vanturnhout@han.nl

L. Munneke
Researcher
University of Applied Sciences Utrecht
Utrecht, The Netherlands
E-mail: l.munneke@hu.nl

Conference Key Areas: Engineering skills, Innovative teaching and Learning Methods, Teaching Creativity & Innovation

Keywords: Pragmatics, Research, Tactic, Inquiry

1 INTRODUCTION

The research for this paper started with our fascination for the pragmatic ways engineering professionals do research, in order to close the knowledge gaps occurring during their problem solving processes. Take for example an engineer who is called to an emergency at headquarters when the internet satellite connection to a remote offshore production plant is down. The engineer can either work around the problem or try to find its cause. Finding the cause of the problem requires thorough investigation on a distant location and it is not clear whether the required spare parts are available. Working around the problem could be faster, but the engineer is not sure if an alternative connection can be created. The engineer decides to probe both ways,

[1] Corresponding Author:
D.W. Greve
dan.greve@hu.nl

analyse the information he receives and subsequently continue on the most promising path. This way, he first succeeds in creating a low bandwidth work-around for the internet connection. Later he finds the cause of the problem so a helicopter can bring replacement parts with the next supplies. (example taken from participant D in this research)

The example illustrates that engineers work in pragmatic ways to close knowledge gaps. They employ pragmatic research tactics that take constrains of time and resources into account and aim for the highest chance to close knowledge gaps. These pragmatic research tactics aim for answers that fit just good enough rather than perfect, to close gaps (e.g. a low bandwidth solution) The answers just need to fit sufficiently in order to solve the problem. To find answers, an engineer can flexibly choose whether to simply look up an answer, investigate the situation more thoroughly or design a rigorous research plan. In summary we define that pragmatic research tactics aim for the highest chance to find answers that fit sufficiently to close knowledge gaps in order to solve the problem with optimal use of time and resources.

As research methodology teachers in engineering education, we would like to have a meaningful discussions based on literature with students and teachers about the use of pragmatic research tactics. A first place to search is the literature on (pragmatic) problem-solving methods in engineering. This literature can be found in the form of best practices for an engineering subject (e.g. ITIL, Six Sigma) or can have a more general life cycle approach with recursive (e.g. problem solving cycle), sequential (e.g. V-model) or incremental (e.g. agile) methods as can be found in systems engineering handbooks [e.g. 1, pp. 32–36]. These problem-solving methods include steps that require research, such as determining user requirements, finding an algorithm or testing a proof of concept. But within problem-solving literature, research methods, and especially their corresponding pragmatic tactics, only receive a global treatment.

Another place to search for research pragmatics is the literature on research methodology in engineering (e.g. [2]–[5]). However, research pragmatics are a topic of interest in most academic textbooks, they are seldom presented as a central concern. And when they are treated as in [6]–[8], pragmatics are discussed with respect to a particular research practice and with corresponding pragmatics in mind. So, although this literature might for example treat how to deal with small samples or missing data, it does not provide help for switching dynamically between small scale or informal research to more thorough approaches.

All in all, the literature on problem-solving and research methods richly supplies solid research strategies suitable to plan research in various types of projects. However, literature on flexible pragmatic research tactics suitable to discuss ways to adapt to the changing situation within projects, is rare and scattered.

To learn more about pragmatic research tactics we set up an empirical study with novice engineering professionals because that is the aspiration level of our students. Since the authors work in bachelor of IT engineering education, the scope was restricted to this area. This resulted in the following research question for this study:

What are pragmatic research tactics that novice bachelor of IT engineering professionals use to acquire sufficiently good answers to close the knowledge gaps that occur in the context of their project assignments?

2 METHODS

2.1 Design

In order to research what novice engineers actually do when closing knowledge gaps, a qualitative approach based on grounded theory [9] was chosen. A concern with the data collection was that part of the process happens invisible and even unconsciously. This entails that written research reports and observations will keep aspects of the process out from view. An interview is a way to collect more anecdotal stories that include part of the unconscious process. Another way to stimulate this is to visualize the process with timeline mapping [10]. To create a rich source of information, the two were combined in a semi-structured visualization interview. The interviews were analysed according to grounded theory, starting with open coding to organize the raw data into labels, followed by axial coding to identify research activities in the labels. The process was concluded by selective coding to find integrating categories of pragmatic research tactics. More explanation of the process, in the paragraphs below.

2.2 Participants

In order to obtain information concerning the research pragmatics of novice IT engineers, interviews were conducted with professionals who have a working experience of three to five years. Within this timeframe, engineers are expected to work professionally and be able to discuss their ways of working in a reflective way. At the same time, they are not yet promoted to senior jobs with different characteristics. Bachelor alumni from the IT program of the University of Applied Sciences in Utrecht were contacted by e-mail. From the positive responses, engineers out of the four IT specializations were incorporated into the study. In this way a broad spectrum of engineers ranging from Technical Software Engineers (TSE), System & Network Engineers (SNE), Software & Information Engineers (SIE) and Business & IT Management (BIM), were included in the study. See Table 1.

Table 1. Participants that were interviewed. See text for explanation of specializations.

Participant	Specialization	Years' experience	Job Title	Practice
A	TSE	4	Test lead medical applications	Multinational
B	TSE	3	Application developer	Own business start-up
C	TSE	5	Product owner	Start-up
D	SNE	4	Network engineer	Offshore industry
E	SNE	5	Network engineer	System consultancy
F	SNE	4	High end network developer	Internet exchange
G	SIE	5	Web developer	Own business
H	SIE	5	High end AV app developer	Secondment, Broadcast company
I	BIM	4	Business analyst	Major consumer web shop
J	BIM	3	Business IT consultant	Development & consultancy
K	BIM	5	BI data warehouse designer	Development & consultancy

2.3 Visualization interviews

For the semi-structured interviews, a form of timeline mapping was used. The participants were asked to reconstruct a recent project by visualizing the problem-solving steps of the timeline with sticky notes and tell stories around the way questions were raised and answered (see Fig. 1a). The sticky notes acted as a visual aid to enable the conversation to switch back and forth between the different problem-solving steps. Since not all research is done as part of a project, the respondents were also asked about the way general knowledge issues were raised and answered in their organization. The face-to-face interviews were conducted by two researchers and took around 90 minutes.

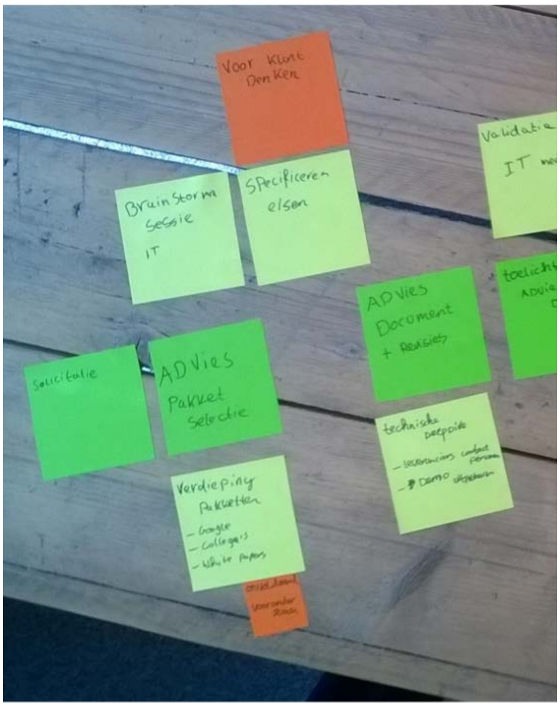
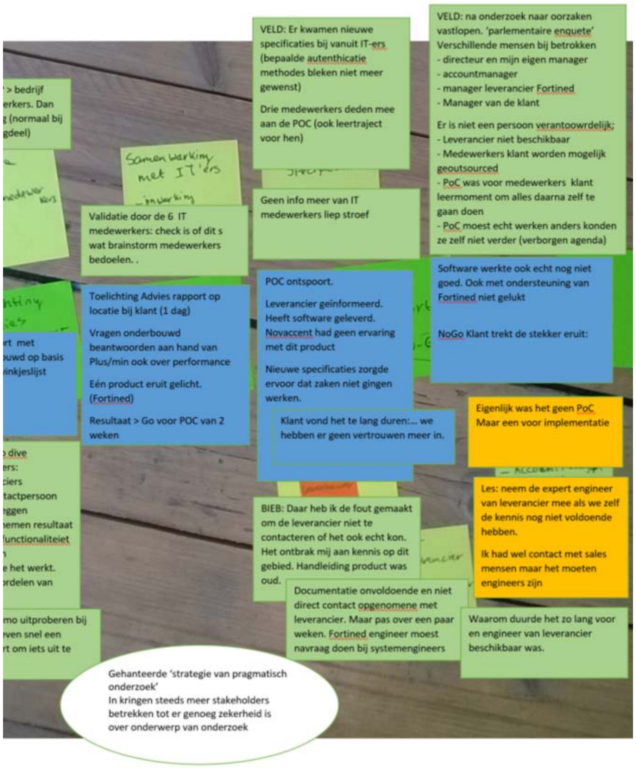


Fig. 1a. Visualization of the project. The sticky-notes were written by the participant during the interview. From left to right in darker green the problem-solving steps in time. To the top and bottom sticky-notes referring to research and knowledge. (Participant E)



VELD: Er kwamen nieuwe specificaties bij vanuit IT-ers (bepaalde autenthicatie methodes bleken niet meer gewenst)
Drie medewerkers deden mee aan de POC (ook leertraject voor hen)
VELD: na onderzoek naar oorzaken vastlopen. 'parlementaire enquete' Verschillende mensen bij betrokken
- directeur en mijn eigen manager
- accountmanager
- manager leverancier Fortined
- Manager van de klant
Er is niet een persoon verantoowrdelijk;
- Leverancier niet beschikbaar
- Medewerkers klant worden mogelijk geoutsourced
- PoC was voor medewerkers klant leermoment om alles daarna zelf te gaan doen
- PoC moest echt werken anders konden ze zelf niet verder (verborgen agenda)
Samenwerking met IT'ers
Validatie door de 6 IT medewerkers: check is of dit s wat brainstorm medewerkers bedoelen. .
Geen info meer van IT medewerkers liep stroef
Toelichting Advies rapport op locatie bij klant (1 dag)
Vragen onderbouwd beantwoorden aan hand van Plus/min ook over performance
Eén product eruit gelicht. (Fortined)
Resultaat > Go voor POC van 2 weken
POC ontspoort.
Leverancier geïnformeerd. Heeft software geleverd. Novaccent had geen ervaring met dit product
Nieuwe specificaties zorgde ervoor dat zaken niet gingen werken.
Klant vond het te lang duren:... we hebben er geen vertrouwen meer in.
Software werkte ook echt nog niet goed. Ook met ondersteuning van Fortined niet gelukt
NoGo Klant trekt de stekker eruit:
Eigenlijk was het geen PoC Maar een voor implementatie
Les: neem de expert engineer van leverancier mee als we zelf de kennis nog niet voldoende hebben.
Ik had wel contact met sales mensen maar het moeten engineers zijn
BIEB: Daar heb ik de fout gemaakt om de leverancier niet te contacteren of het ook echt kon. Het ontbrak mij aan kennis op dit gebied. Handleiding product was oud.
Documentatie onvoldoende en niet direct contact opgenomene met leverancier. Maar pas over een paar weken. Fortined engineer moest navraag doen bij systemengineers
Waarom duurde het zo lang voor en engineer van leverancier beschikbaar was.
Gehanteerde 'strategie van pragmatisch onderzoek'
In kringen steeds meer stakeholders betrekken tot er genoeg zekerheid is over onderwerp van onderzoek

Fig. 1b. Open coding labelling of the interview, based on visualization and the recording, by one of the researchers. From left to right in blue the labels for the problem-solving steps. To the top and bottom labels referring to research and knowledge.

2.4 Analysis of the interviews

Each interviewer analysed half of the interviews based on the visualization and the recordings in a process of open coding to create labels [9]. This resulted in a detailed description of the process steps of the project in the visualization (see figure 1b.). To create uniformity and mutual understanding the other interviewer checked and completed the work and vice versa. Based on this detailed description of the visualization, labels about research activities were identified in a process of axial coding [9]. This resulted in the description of 77 research activities that were placed in Excel by one of the researchers and checked by the other. The quotes on the research activities were split in accordance to the approach en the sufficiency of the research activity. Below an example of a research activity from participant A:

Research approach: *Before we commit software it is always reviewed. Not so much because the code gets tidier (also happens) but especially that you are together and the other person knows what you did.*

Research process sufficient when *: I do not like that piece of code because ... and then the other person says: yes I did that because ... and then you say: oh yes. What is important in a review is human contact. Do not crack down on someone else's code with a tool without contact.*

During and after this process of identifying the research activities, several integrating categories of research activities were defined in a process of selective coding [9]. The resulting categories were subsequently contributed to the research activities after mutual understanding between the two interviewers. The integrating categories are described in the results section below.

3 RESULTS

In the introduction of this paper, we defined that pragmatic research tactics aim for the *highest chance to find* answers that *fit sufficiently* to close knowledge gaps in order to solve the problem with optimal use of time and resources. This section will describe what pragmatic research tactics were found with selective coding in the research activities of the participants. Firstly, we will describe three categories of tactics that aim for the highest chance to find answers. Secondly, three categories of sufficiency to determine whether answers fit sufficiently to close the knowledge gap.

3.1 Tactics

In the research activities, three different tactics could be identified that aim for the highest chance to find answers: concentric, iterative and probe-response. A research activity is categorised as **concentric** (found in 21 research activities in 11 projects) when it contributes to research that starts with sources of information directly accessible to the researcher and the circle of possible sources is widened until the answer is sufficient. An example from participant E is a brainstorm meeting with clients from the IT department to specify a new system. These specs were the starting point for the development and were later validated and supplemented by a group of users.

A research activity is categorised as **iterative** (22 activities in 8 projects) when it contributes to research that starts with an answer that bridges a basic part of the knowledge gap. If necessary, the knowledge gap can be closed more thoroughly in an iterative way. An example is a research activity of participant F. He works as a network engineer for a big internet exchange provider. As part of a project, he had to test network equipment that was merely on the market and uses innovative fibre technology. He created a basic test setup and asked several suppliers for a demo with the test setup. With every test demo, he accumulated more knowledge about the technology and the test equipment until he had a test setup that met the highest industry standards.

A research activity is categorised as **probe-response** (7 activities in 5 projects) when it contributes to research that starts with an educated guess (probe) and draws upon the received response whether the result is satisfying or requires a new probe. An example is a research activity of participant G. He had to select a pdf generator for the web application he was building. He assumed this was standard technology and looked for commonly used solutions on Google, picked one, tried it for five minutes, found some usability issues he did not like, tried the next, and found it more satisfying. He knew this was not perfect either, but he could handle it so he continued to use it in the application.

It is relevant to mention here, that the research activities of all 11 projects show that the research tactics are not always matching with the problem solving strategy of the project as a whole. In 34 activities in 10 projects the research tactic and the problem-solving strategy of the project match each other’s respectively concentric/recursive (e.g. problem-solving cycle) or iterative/incremental (e.g. agile) nature [1]. But in 20 activities in 11 projects, they do not have the same nature. Participant H for example works in an environment with an incremental iterative problem-solving strategy (Scrum). But he uses a concentric research tactic when he is not able to find programming solutions ‘*as rule we ask for help in our team if we get stuck for more than 30 minutes. Often a team member knows the answer, if not, we move on to support groups*’.

3.2 Sufficiency levels

The first paragraph described three tactics that aim for the highest chance to find answers. But when can an engineer stop the concentric cycling, the iterations or the probing? When does the answer fit sufficiently to close the knowledge gap in order to solve the problem? The results show that this depends on the ambition of the project and that the research activities could be categorised into three levels of sufficiency: check for viable answers, boost critical demand and change the game.

The minimal sufficiency level that could be found were research activities with answers that just fit sufficiently to make the solution viable. These activities were categorised as **check for viable answers** (found in 34 research activities in 11 projects). The 30 minute rule to call for help, described in the paragraph above, is an example of looking for this minimal level of sufficiency.

The following level of sufficiency was categorised as **boost critical demand** (24 activities in 9 projects). At this level participants looked for critical demands in the project and put extra effort in the related research activities to find a major improvement. The remaining research activities of these projects typically showed a *check for viable answers* sufficiency level. This indicates that the *boost critical demand* level **works on top of a *check for viable answer*** level. The *boost critical demand* level aims at solutions that are considerably better than prior solutions. An example can be found in a research activity of participant J who had a really tough issue with response times for an application. In consultation with the client they directed significant project resources to this problem. Interestingly they used a *probe-response* tactic because they did not start with weeks of diagnostic research but picked a promising part of the problem and tried to solve it with trial and error. After a couple of days it turned out this did not work so they stopped and inspired by the results picked another promising part for a next series of trial and error etc. After two weeks the delay was reduced to 90% and the client was satisfied.

The highest level of sufficiency was categorised as **change the game** (8 activities in 3 projects) and found in projects were engineers looked for new avenues to gain a competitive advantage. In these projects several research activities were only sufficient if the solution changed the game. Other activities in these projects could have a *check* or a *boost* level. A *change the game* example is participant F who works as a network engineer for a big internet exchange provider. The project assignment was to test the feasibility of using equipment that was merely on the market with Dense Wavelength Division Multiplexing on colored lasers and advanced forms of modulation (QWAM) in order to significantly reduce setup time for new fiber connections. He applied a concentric research tactic to find the desired information. He started with product specifications and went on to the white papers from the manufacturers, all the way into the original scientific articles. Later he applied an iterative tactic to create a test setup that met the industry standard (see example in paragraph on iterative tactics). The bar was raised to the highest possible level and the solutions were only sufficient when all criteria were met in the test setup. Only one of the world top suppliers succeeded.

Table 2. Examples of Sufficiency levels of Tactics for Pragmatic Research Tactics (Participants are represented by letters A…K. See table 1 for explanation)

Pragmatic Research Tactics				
Sufficiency level Pragmatic		**Check for viable answers**	**Boost critical demand**	**Change the game**
Tactics	**Concentric**	G: Satisfying solution + one quick win = OK. E: Validation of user requirements from brainstorm with 6 users. A & H: Do not mess around, ask questions in team: if quick answer fine; else move on.	G: Solution should work fluent on graphically limited screens. C: Backend that is uniform and scalable for all our client applications. I: Priority workshop with UX, BA, Search and product experts.	F: All specifications are grounded in peer reviewed papers. F: Formulate extremely high functional and maintenance requirements; Design a lab test; No compromise, only solutions that pass all tests are accepted.
	Iterative	A: Automation of smoke test (daily test of new code) as starting point for continuous integration. C: Minimal Viable Product. G: Until it works and I dare to take the next step.	A: Developers use test automation as part of daily routine for continuous integration. J: Risky Assumption Test: early checks with customer. Learning whether or not you are wrong as early as possible.	A: Test automation program is accepted as evidence for our medical application by very critical Food and Drugs Authority (FDA) in USA. (which would take away an enormous load of paperwork in our organisation)
	Probe-response	D: Temporary solution works (low bandwidth internet in example from introduction).	J: (critical) delay in application was reduced with 90% and acceptable for client.	No example from the participants in the data.

4 CONCLUSION AND DISCUSSION

The results show that potentially valuable contributions to the literature are pragmatic research tactics that consist of:

- Three different tactics to aim for the highest chance to find answers: *concentric, iterative* and *probe-response.*
- Three levels of sufficiency to close knowledge gaps in order to solve the problem at hand that correlate with the ambition of the project: *check for viable answers, boost critical demand* and *change the game*

Furthermore, the results reveal that in all the projects the research tactics do not always match with the problem-solving strategy of the project as a whole. This strongly indicates that pragmatic research tactics actually have a distinctive level in the whole problem-solving process. Additionally, we can conclude from the results that the three tactics have a cumulative information gathering approach in common. These cumulative tactics make it possible to move flexibly up and down in what can be called the *pragmatic research spectrum* of simply looking up an answer, investigating the situation more thoroughly or researching it in a planned and rigorous way.

With the results of this study, it should be possible to make students and teachers aware of pragmatic research tactics and discuss the use of them in a meaningful way. Starting for example from the ambition of the project, the sufficiency level of different research activities can be discussed. Subsequently, the choice of a pragmatic tactic can be deliberated and the chance it provides to flexibly close the knowledge gap, with optimal use of time and resources in mind. Furthermore, the results provide various examples to illustrate the abstract parts of the discussion and make it practical.

Although the research has focused on IT engineering, it seems reasonable to expect similar pragmatic tactics in other engineering practices.

Finally, this research of course has several limitations that leave room for further research. One aim should be to validate whether and to what extent the tactics and sufficiency levels can be found in other (IT) engineering projects. Another interesting subject of research is to determine how engineers (can) make the right choices in de use of pragmatic research tactics and how engineers can perform them properly.

REFERENCES

[1] Walden, D. D., Roedler, G. J., Forsberg, K. J., Hamelin, R. D., & Shortell, T. M. (Eds.). (2015). *INCOSE Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities, 4th Edition*. Wiley, Hoboken, New Jersey.

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

[3] VanTurnhout, K., Craenmehr, S., Holwerda, R., Menijn, M., Zwart, JP, and Bakker,R. (2013). “Tradeoffs in Design Research: Development Oriented Triangulation,” Proc. 27th Int. BCS Hum. Comput. Interact. Conf., pp. 1–6

[4] Oates, B. J. (2005). *Researching information systems and computing*. Sage. London.

[5] Vaishnavi, V. K., & Kuechler, W. (2015). *Design science research methods and patterns: innovating information and communication technology*. Crc Press, Boca Raton.

[6] Knight, P. T. (2001). *Small-scale research: pragmatic inquiry in social science and the caring professions*. Sage, London.

[7] Layder, D. (2012). *Doing excellent small-scale research*. Sage, London.

[8] VanTurnhout, K., Holwerda, R., Jacobs, M., Bakker, R. (2015). “Innovatie, een zaak van pragmatiek,” Informatie p10-17

[9] Strauss, A., & Corbin, J. M. (1990). *Basics of qualitative research: Grounded theory procedures and techniques.* Sage, London.

[10] Kolar, K., Ahmad, F., Chan, L., & Erickson, P. G. (2015). Timeline Mapping in Qualitative Interviews: A Study of Resilience with Marginalized Groups. *International Journal of Qualitative Methods*, *14*(3), 13–32.