The Triple Schizophrenia of the Software Engineering Researcher [*]

David Benavides, Antonio Ruiz-Cortés, and Carlos Müller

Dpto. de Lenguajes y Sistemas Informáticos
University of Seville
Av. de la Reina Mercedes S/N, 41012 Seville, Spain
{benavides, aruiz, cmuller}@tdg.lsi.us.es

Abstract In this paper we question the problem of a software engineering researcher, who in his daily work, has to deal with researching, teaching and learning activities at the same time. Likewise, we suggest the Action Research as the way to disentagle from that triple schizophrenia.

1   Introduction

Once upon a time in any university, a software engineering student. This person decided to study engineer science instead of mathematical or physics, mainly because he liked to build things. It turns out that our friend step by step, was realizing that his professors came from different branches of knowledge and they assure him that their subjects were very important to his future engineer's career. He, convinced about it, did not stop studying but was starting to realize about the time was spending and the principal reason to have chosen engineer as his career did not appear. The few ones that teaching about constructing things, were not doing it as well as our character demanded because the time dedicated to these subjects were really insufficient. Moreover there were many teachers that did not seem to really know how things are constructed. Finally, that student ended the university studies and thankfully he had a gang of friend with the same worries he managed to get enough knowledge to finish his formation in a good level. Otherwise he would not had known how things are constructed. This student worked for a while at different companies and noticed that in addition to constructing things he also liked researching to figure out the best way of constructing these things and, because he heard about the university as the best place for researching, he ended being employed as a lecturer at the university where he studied. After a few months of adaptation, when he started to know what was researching and what was a doctoral dissertation in software engineer, he was surprised on recognize that many of the doctors in his department did not realize well enough these concepts. He had friends researching in mathematics, physics or even industrial engineering that knew it well and he started to become worried. On the other hand, he started to teach matters which he had to prepare quickly due to finishing touches in the syllabus he has no time to prepare. He

realized that to speak in public in front of three hundred pupils was not the same that to speak to a business client. At the same time he saw as the pupils were demanding, in the same way he did when he was a student, knowledge on how to construct software and in addition they wanted that this knowledge was accompanied of cutting-edge technology, for what he had to start studying in order to prevent his contents become obsolete. After few years, one day, he discovered that in his work he had to play three roles that for different circumstances he wanted to fulfill by the best way: he was, in one side, professor and he had to do it well, and on the other hand he was a researcher and on which depended his career and at the same time he had to be a student to learn in two directions: one to up-to-date his teaching knowledge and another to up-to-date his knowledge on researching. Many of his colleges abandoned some of the roles to focus on others, many other fell down in the gap of the triple schizophrenia of being a researcher, professor and student at the same time, so he realized on time and without leaving aside any of the planes, it is not known how, he was saved.

This fictitious history can reflect the life of many researchers in software engineer. Research in software engineer can be considered relatively a new discipline compared to others more consolidated like mathematics, physics or even computation sciences.

At the university, a big number of researchers live in the constant schizophrenia of being a good professor and researcher (the emotion of schizophrenia of a university researcher was already described by Ortega y Gasset [12]). In a new field and constantly changing as is software engineering, this schizophrenia is joined to the fact of being also a good student with the aim of being in a constant education and don't fall away from the train of technological innovation. In the daily work of a software engineer researcher are mixed some disciplines so distant and different like computational sciences (formal level), technological innovation (learning level) and pedagogical aspects (teaching level). To get out of this isolation and anguish emotion where the researching can fall into due to the triple schizophrenia, we consider necessary the following: first, change gradually the lines over where we have to work (what could generate a wide and long debate where we think we can opine but not suggest), and second, draw a path where we can walk through without leaving aside any of the of the levels of the triple schizophrenia, for what we foresee that the "action research" could be a possible path that take the software engineering researcher out of the sensation of isolation.

2   The Schizophrenia Teaching-Researching.

There are areas of the knowledge where researching and teaching can go together. However, it is well known that other areas of engineering there is a difficult relation between teaching and researching. Researching and teaching are duties that require different skills and pursue different goals. Let's propose as an example the following differences established by Felder[3]: i) whereas the main goal of researching is to give new knowledge, teaching has as objective to settle such knowledge (normally far from what is researched) and contribute with enough capacity to solve real problems. ii) the verbal skills can be useful when researching without being a required tool, while there are essential for a good teaching task.

The duality teaching-researching becomes more complicated for the fact that in the software engineering field there is still a huge debate on the nature of the term, the nature of the discipline and whether to consider software engineering as a science or as a technology [7,4,5,9]. Therefore, trying to carry on researching and teaching works in a high level is, almost always, something even more incompatible with the reforms that are coming to the European educational system (Bolonia). On the other hand, bet for one of them will make the other to lose quality. In consequence, we consider suitable to pose the next question ¿teaching or investigation?. In the Spanish University, the first point is the reason why researchers are paid (the positions' name are associated professor, collaborator, full time professor, etc) and therefore, from a ethic and utopist point of view, it should be the most time-consuming task, reducing, as a result, the researching capacity. However, the second leads you to promote and this is what professors use to bet for.

3   One More Level.

There are many areas where the researcher can encounter the constant struggle between being teacher and researcher, however, in an area like software engineering where changes come up almost every second, there is an additional level that raise the difficulty of matching the duet teaching-researching: the learning level. It is supposed that a good researcher has to have enough learning capacity and at the same time, it is supposed that a software engineering professor has it as well. However, if we join a researcher and a software engineering professor in the same person (which occurs almost always) we will find that the big amount of information that would be necessary to assimilate in order to feed the researching (formalisms, languages of specification, computational theories, mathematical notations, theorems and theories, etc.) is not too useful in teaching tasks, and vice versa. Trying to re-use what we research for teaching tasks can lead us, in many occasions, to diminish the interest of what is being taught and go far from the goal of the student protagonist of the story.

4   Action Research as a potential antidote.

In order to prevent that the sensation of isolation that can cause to the software engineering researcher make him smother for the triple schizophrenia, we believe that are three possible means:

1. Abandon any of the levels: i) leave aside teaching which we think is not ethical and little beneficial for our end clients (students and society), ii) leave aside researching, which will rebound negatively in the researcher itself and could leave aside what society expects from us, iii) leave the constant up-to-dating, what for sure will reduce the quality in the two aspects before mentioned. 2. As Felder suggested [3], redefine the structure and roles in order to separate the teaching duties from the researching ones, separating a group of people dedicated to a specific task and other group to another, regardless the possible synergies that they can find between both groups of people and activities. Or even more, to introduce in the faculty the concept of specialization in such a manner that it could allow to take advantages of the virtues of each person in the task

he is more efficient. 3. Try to find a shortcut that makes that the three levels are feed in such a balanced manner that permits to keep in a sustainable balance (recall that what we are trying to avoid is to fall in the craziness of the triple schizophrenia) between teaching, researching and learning.

At this time, we believe that the best way could be to choose this last point. In our experience, we think that the traditional quantitative methods are not useful in the task of getting with the trinomial researching-teaching-learning, what a software engineer researcher has to deal with. Therefore, we believe that we can open the debate to find a method that allows the researcher to cohabit with these three levels in an acceptable quality status.

'Action research' is characterized for focusing on the resolution of real problems out of the researcher's boundaries, serving, at the same time, to feed the knowledge in the field of researching. Consequently, it seems that in order to escape of falling in the craziness of the triple schizophrenia, a possible antidote could be to design an action research plan in software engineering efficiently enough to obtain a level of quality for teaching and researching tasks. In one hand, due to the nature of action research where the resolution of problems is posed as a premise, it would easy and fruitfully reuse this knowledge in teaching duties and, therefore, the effort in learning duties could be shared by both activities. Nonetheless, it is not less true that the design of this investigation plan is not an easy task and the necessary effort to design it can make us fall in the trap of the new schizophrenia.

Our experience could state that in the work of an engineer (not a researcher), many common points exist and we can learn about the solutions gave in his environment. Not researcher engineer, also is in many occasions in this schizophrenic state that in many points looks similar to the state of a researcher. We find many problems that he must settle, for example: he has to take care of his clients and to satisfy their needs with enough quality, it is alike to the work of teaching of the researcher when he is doing teaching works. He must not also neglect any new aspect on his business that grants him major competitiveness in his environment of work, which has certain similarity with the research in the university field. Finally, it might mention that the engineer must realize the mistakes committed by his organization or by the competitors, to not fall down in them and to be in a continue progress process, that might have points in common with this plane of learning of the researcher-teacher indicated previously.

One of the approaches done in production environments, which can be studied by a not researcher engineer to satisfy the schizophrenia is to use the so called Technological Monitoring improved with the Competitive Intelligence that as we are going to see now, might consider really as a plan of Action Research.

5   Technological Monitoring/Competitive Intelligence as a research-in-action plan.

Technical Monitoring (TM), is defined as the capture, analysis, diffusion and exploitation of the technical information useful for the enduring and growing of a company [10].

On the other hand, Competitive Intelligence (CI) consists on transforming this knowledge in intelligence, that is, analyze the information and give it any function. [11,6].

Few authors [10,11,6] have suggested a cyclic process that would integrate both business methods. Figure 1 shows the activities of such joint process. As we can observe, there are five activities sequentially accomplished:

– Definition of goals: what is what I have to search, and where should I find it (information sources).
  The number of goals to fix in this stage must be fit to the necessities of the company, since depending on the characteristics of the sector that covers it, it can vary its priorities in this process of constant improvement of TM+CI, and for instance, if the company belongs to a very changing business, this task will be essential.
  The object of our information pursue is all those sensible information that can make the entrepreneur accomplish any kind of competitive advantage for his business, as could be technological information, new production methods applied to his organization or even productive process already invented but still not applied in our sector. And the source of information can be very diverse, specialized press in his type of business, technological press, reports on new patents with applicability in his business, public information provided by other companies of the same field.

– Gathering of Information: it refers to the works done in the process of obtaining this information. It can be done manually - using documentary makers and infonomist people - or in a semiautomatic way - by searching software, as metaseekers, that permits us to choose the search engines to use, to do summaries of web content, to notify about the changes on web pages, to analyze the results of the search, etc. Margarita Aeste quotes: "Tools in this manner allow not only to find information but also to generate knowledge that take into account the learning style of each person in a way that each one can select which fit better to him" [8].

– Storage: in this stage all the information obtained in the previous phase is gathered and a first classification of data is accomplished. With the aim of, through a correct knowledge management, such information will be saved for a further analysis.
  The classification criterion depend on the type of information compiled, although, drafting a general classification, that data can be external - given by the company and its work methodology - or internals - these could be obtained after a correct knowledge management.

– Analysis: study of such data by filtering and observing those nuances that can be more important for the entrepreneur. This analysis involves the utilization of all data obtained from the previous phase. In addition, the study of the data must be accomplished taking into account the business model and the actuation of the business that is being treated in order to be able to bring complete solutions that makes our company more competitive.

– Adaptation: that analysis so exhaustive and fitted to the global necessities of the institution implied, therefore, a change in the attitude of the company. The analysis show facts that are necessary and essential to achieve competitive advantages over the companies in the same business.

Finally, we should redefine the first-step goals because the technological monitoring is a sequence of repeating activities, therefore, we have to redefine the goals from the current phase of the institution of the client and interest that he had, to restart again from the first step.
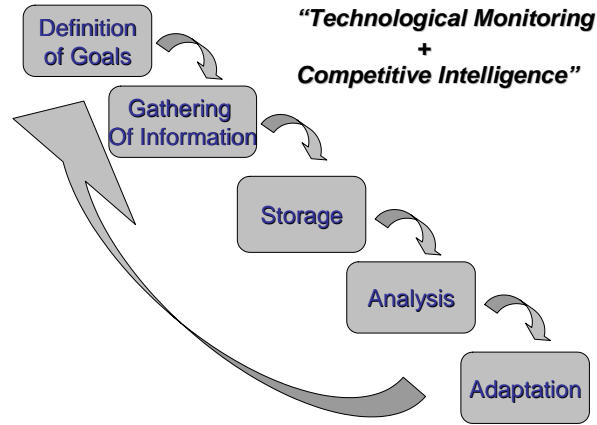

Figure 1. Sequence of Technological Monitoring + Competitive Intelligence

We can see the correspondence between the different stages of each sequence in Figure 3.

We can demonstrate that, although the correspondence between the stages of the two models is not one-to-one, everything that action research stages represent is reflected in one or two stages of TM+CI. As a result, we can conclude that the group made by Technological Monitoring and Competitive Intelligence is a action research model that solves the similar problem of the researcher but in a real problem of the business environment.

Therefore, we think that the process done in VT+IC could be included in Action Research plans, in such a way that it could pacify the effect of the terrible schizophrenia. The software and methods used in VT+IC adapted to the Action Research can be of great utility to fight with the different rolls of the researcher. Thus, by the experience obtained in environments far away from the University and the study of the VT+IC - that is in essence a process of action research- we could try to extract this know how and try to apply in the university scenario, as a possible solution to get rid of the triple schizophrenia that we are treating.
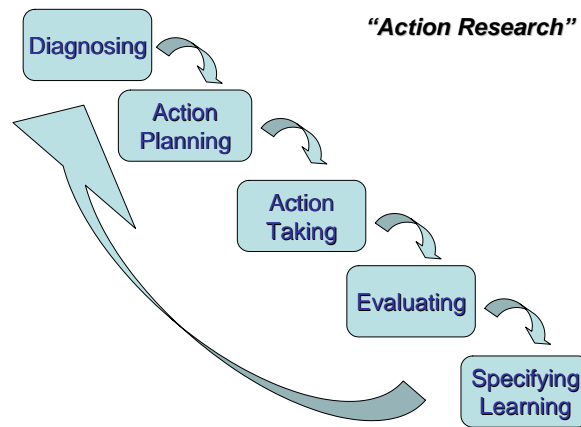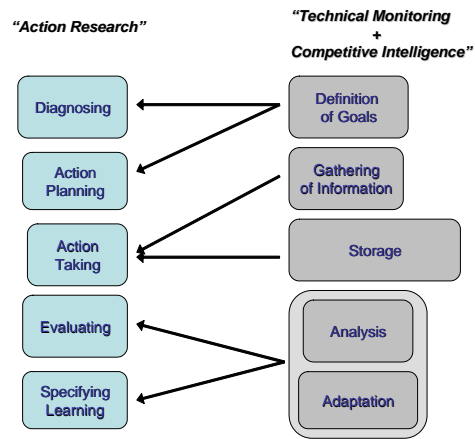
Figure 2. Sequence of action research, [2]

Figure 3. Correspondence between TM+CI/Action Research sequences

References

1. *2º Workshop en: Métodos de Investigación y Fundamentos Filosóficos en Ingeniería del Software y Sistemas de Información (MIFISIS'2004)*. Servicio de Publicaciones Universidad R. Juan Carlos. Madrid, 2004.
2. Richard L. Baskerville. Distinguishing Action Research From Participative Case Studies. *Journal of Systems and Information Technology*, pages 25–45, 1997.
3. R. Felder. The Myth of the Superhuman Professor. *J. Engr. Education*, 2(82):105–110, 1994.
4. J.M. Cañete F.J. Galán. ¿qué se entiende, en españa, por investigación en ingeniería del software? In *MIFISIS*, 2002.
5. V. Guijarro. *¿Crisis del Modelo Lineal? Aproximación a Propuestas Alternativas en Filosofía de la Tecnología*. MIFISIS, 2002.
6. Alessandro Comai Joaquín Tena Millán. Formación en Inteligencia Competitiva. *PUZZLE -Revista Hispana de Inteligencia Competitiva*, 3(12):4–9, 2004.
7. E. Marcos. Investigación en ingeniería del software vs. desarrollo software. In *MIFISIS*, 2002.
8. directora del Área Educativa de QUIPUS Margarita Aeste. ¿Cómo podemos visualizar información? *PCWORLD (México)*, 2004.
9. D. Parnas. Software engineering: an unconsummed marriage. *Communications of the ACM*, 40(9), 1997.
10. Ramón Maspons Pere Escorsa. *De la Vigilancia Tecnológica a la Inteligencia Competitiva*. Prentice Hall, Colección Financial Times,. Madrid, 2001.
11. Puzzle04. La Inteligencia Competitiva en las mejores prácticas españolas. *PUZZLE -Revista Hispana de Inteligencia Competitiva*, 3(13):25–28, 2004.
12. Ortega y Gasset. Misión de la Universidad. *Editorial Revista de Occidente*, pages 15–78, 1930.