Understanding Requirements Engineering: a Challenge for Practice and Education

Lemai Nguyen
School of Management Information Systems
Deakin University
Vic Australia
lemai@deakin.edu.au

Jocelyn Armarego
School of Engineering
Murdoch University
WA Australia
jocelyn@eng.murdoch.edu.au

Paul Swatman
Stuttgart Institute of Management and Technology
Germany.
swatman@uni-simt.de

Abstract

Reviews of the state of the professional practice in Requirements Engineering (RE) stress that the RE process is both complex and hard to describe, and suggest there is a significant difference between competent and "approved" practice. "Approved" practice is reflected by (in all likelihood, in fact, has its genesis in) RE education, so that the knowledge and skills taught to students do not match the knowledge and skills required and applied by competent practitioners.

A recent action research study describes a new understanding of the RE process. RE is revealed as inherently creative, involving cycles of building and major reconstruction of the models developed, significantly different from the systematic and smoothly incremental process generally described in the literature. The process is better characterised as opportunistic and insight driven.

This mismatch between approved and actual practice provides a challenge to RE education - RE requires insight and creativity as well as technical knowledge. Traditional learning models applied to RE focus, however, on notation and prescribed processes acquired through repetition. We argue that traditional learning models fail to support the learning required for RE and propose both a new model based on cognitive flexibility and a framework for RE education to support this model.

1. Introduction

Requirements Engineering (RE) has often been widely recognised as a cognitive process, involving conscious mental efforts to come to a decision (see for example [3, 29]). Guindon [17], in addition, argues and stresses that the problem solving activity in RE should not be treated as being "*preceded by an independent and complete phase of problem understanding*" [17, p.729]. In fact, the RE process should be described more completely as involving both problem understanding and problem solving activities inter-twined.

Furthermore, RE is complex and knowledge intensive [41], dealing with the conceptual complexity and wickedness of the requirements problem [5]. The term "wicked" was coined by [41] to describe problems in general policy planning which cannot be definitely described, they have no stopping rule and no ultimate criteria for the evaluation of solutions, they are unique and involve pluralistic perspectives of the participants. In RE, the wickedness (ill-structuredness) is defined as the incomplete and ambiguous representation of problems, the multi-discipline domains and knowledge, the non-deterministic approach to solving requirements problems and the open-ended nature of solutions [18, 3].

Clearly, RE bears many similar characteristics and properties of general design and problem solving activities, such as the wickedness of the problem, the complex cognitive process and the management of different sources of knowledge involved, and creativity required [41, 45, 27, 49]. It would be sensible to expect that theories related to the general design and problem solving process described by various authors, such as [27, 49, 44], would also apply to understanding the RE process.

Over the last decade, many authors attempted to describe the structure of the RE process and what the requirements engineers (REers) do during this complex, cognitive and knowledge intensive process. There is some variation, in the literature, in the descriptions of the RE process. Traditionally, the process is seen as hierarchically organised, with the decomposition of complexity into smaller, manageable units. Recently, opportunistic behaviours are observed to be critical in the RE [18, 24, 6]. However, the questions of how they occur, what their triggers are, and what their impact on the complexity of the problem and the requirements model is, have not been described adequately.

Moreover, reviews of the state of the professional practice in RE stress that the RE process is both complex and hard to describe, and suggest there is a significant difference between competent and "approved" practice. "Approved" practice is reflected by (in all likelihood, in fact, has its genesis in) RE education, so that the knowledge and skills taught to students do not match the knowledge and skills required and applied by competent practitioners.

This paper attempts to address this mismatch by describing a new understanding of the RE process, revealed in a recent action research study, and discussing its implications in RE education. The paper is structured as follows:

- Section 2 briefly describes a new understanding of the RE process.
- Section 3 discusses its implications in teaching and suggests a new learning model and a framework for RE education to support the model.
- Section 4 concludes the paper and outlines a plan for future research.

2. Catastrophe-cycle RE process

Design explanation[1] is information which represents and explains the reasoning behind the design process. Our critical review of the RE and design explanation literature suggests that design explanation can provide the requirements engineer and the project manager with many potential benefits. Therefore, we set out to explore and investigate the use of design explanation in understanding and monitoring the RE process. In pursuing this overall objective, we undertook an in-depth study into understanding the process of RE.

This paper focuses on describing the new understanding of the RE process gained from the study. More details about this project can be found in [34, 36].

2.1. Research method

The research method adopted was action research. Hermeneutics cycles of action research allowed the research ideas/concepts to be generated, refined and evolve. The active and reflective characteristics of action research enabled the researcher to experience the development process and reflect upon her actions. In addition, the researcher was able to interpret the story of what was actually happening during the project, i.e. not relying on retrospective interviews or the accuracy of research subjects' explanation of what they did, and how and why they did it.

On the other hand, the limitations of action research are the restriction to the difficulty in generalisation of the results, the inability of the researcher to be unbiased, and possible different explanations by the researcher of events [1, 12, 13].

The real world situation for the action research was a RE project in Australia. The project involved 3 participants─all of whom were experienced REers─and an expert in design explanation. The project and took place over a long period of 18 months. All the participants have degrees in Computer Science and Information Systems and all have industry experience. Our findings from this project also are confirmed by their expertise in problem understanding and solving.

The project involved the development and specification of requirements for a CASE tool to support FOOM [47], a RE method. It is important to recognise that the most challenging aspect of this project was that of requirements engineering, not solution design.

The research data include both requirements documents and the RE process:

- Intermediate versions the requirements model at different development stages
- The decision record with the underlying deliberations
  - what decisions were made
  - when they were made
  - why they were made
  - what contextual information (i.e. about the decision and its rationale) was taken into account in making the decision

IBIS, an *ad hoc* approach to design explanation [10] was used in capturing and recording the decisions as they were made. The IBIS documents were attached to their associated intermediate versions of the requirements model. Occasionally, QOC, a *post hoc* approach to design explanation [30] was used in reviewing, supplementing, and reorganising the IBIS base.

Snapshots of the requirements specification, IBIS documentation and QOC reviews form the primary basis for our analysis of the requirements process. The observation data were qualitative and semi-structured. The research data provide us with a record of the dynamics of the requirements model, the underlying deliberations made during the entire RE process and the experience of the researcher during the course of action. This paper briefly describes the findings drawn from our interpretive analysis. Detailed explanations and justifications of the findings can be found in [34, 35, 36].

[1] Design explanation is also referred to as design rationale in the literature. In our approach, design rationale is used with emphasis on explanation.

2.2 Research findings

The catastrophe-cycle RE process

Overall, the RE process was cognitive, involving continuous mental effort to understand the requirements problem and to make decisions to solve it. The data show frequent opportunistically guided episodes of a range of activities: acquiring and understanding information from problem domains and representing and validating it in the requirements model. This somewhat chaotic process reflects a normal RE practice and also confirms the description of these cognitive behaviours offered in the literature [3, 46, 7].

A close examination of the research data shows that the RE process is inherently creative, involving cycles of building up and major reconstruction of the requirements model, significantly different from the systematic and smoothly incremental process generally described in the literature. We characterise the pattern of construction and reconstruction of the requirements model through our catastrophe-cycle RE process model (see Figure 1).

Figure 1. The catastrophe-cycle RE process

Our initial observation and examination of the decision records and the intermediate versions of the requirements model captured shows that:

- As the requirements were acquired, analysed, and added into the requirements model, the complexity of the model grew over time. At some critical point, the requirements model was significantly simplified and restructured. The problem space was reconceptualiased and the requirements model had a new architecture and reflected a new perception of the requirements problem by the requirements engineer.
- The critical points happened as a result of a sudden, unexpected insight, i.e. not through systematic and planned efforts.
- Although the overall complexity of the model was reduced significantly at the crisis points, our learning about the requirements *must* have been increased. For example, the requirements were better understood and/or solved from a new perspective (often with a more holistic view) and the problem space was more elegantly structured. It became clearly evident that the relationship between model complexity and level of understanding is not monotonic.

Essential, incidental and accidental complexity

Our initial observations led to a deeper examination of the dynamics of the complexity in the requirements model. As a result, we were able to explain catastrophe-cycle RE process pattern through the lens of three different types of complexity[2] in the requirements model. Rather than viewing the complexity of the model as a single value which indicates the structural size of the requirements model, we distinguish the following types of complexity:

- *Essential* complexity represents the intrinsic understanding of the requirements problem gained and embedded in the requirements model. This type of complexity grows over time towards the "completeness" of the problem complexity as we can assume that our understanding of the problem must not be decreasing.

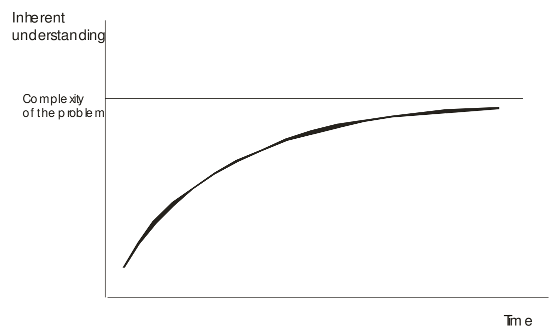

Figure 2. Essential complexity

- *Incidental* complexity represents the complexity of representation rather than substance in the model. In other words, it shows the poor fit between the structure of the requirements model and the structure of the real world problem that the model attempts to mirror. It grows as the requirements model develops and new

[2] All the discussions and diagrams in this paper were based on our qualitative analysis and interpretation of the RE process recorded. Although there is a strong need for quantitative measures to support monitoring of the complexity of the requirements model, selection or development of an RE metrics scheme is not straight forward and is the subject for further research.

requirements are added to the model. The more components are added into the model, the more difficult it becomes to add new components into the existing increasingly complex model. Therefore, the incidental complexity grows exponentially over time.

At crisis points, when the requirements problem is reconceptualiased and the requirements model is restructured, the incidental complexity is reduced significantly.

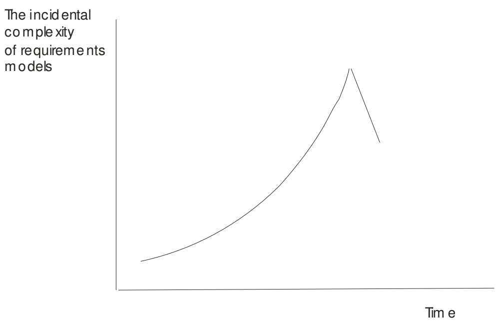

Figure 3. Incidental complexity

- *Accidental* complexity represents the hidden knowledge in the requirements model which becomes explicit only as a result of reconceptualisation insight at the crisis points. After the model is restructured, the accidental complexity becomes a part of the essential complexity which continues to grow over time.

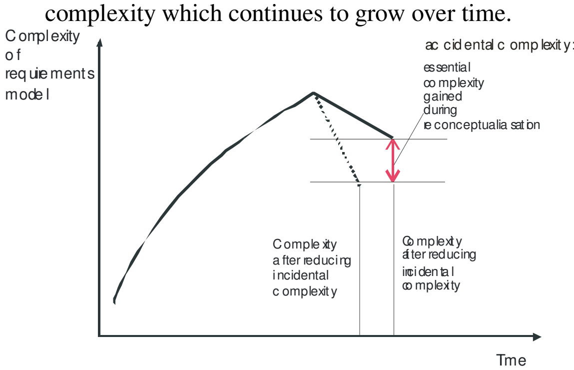

Figure 4. Accidental complexity

The catastrophe cycle RE process can be explained in more detail using the dynamics of these types of complexity as follows[3]:

- As time progressed, requirements are gathered, clarified and analysed, the problem is explored and structured. As understanding of the problem gets mature, partial problems are explored, solved and, in turn, trigger further problems. The problem space is continually expanded with new directions being revealed, investigated and structured. This is consistent with Visser's [49] argument that, with ill-defined problem situations, designers *construct* the problem space.
- Working on a problem area often requires the problem solver to revisit previous solved and/or partially solved problem areas. Indeed, our decision record shows that the path that led from one problem area to another was rather opportunistic and unpredictable. While this is different from the systematic decomposition approaches described in the literature (for example [22, 21], it is consistent with the opportunistic characteristic of the design process described by Schön [44, p.175]: "*As you worked on a problem you are continually in the process of developing a path into it*". In RE, this is supported by Khushalani [24] and Carroll and Swatman [6].
- As a result of the exploration and modelling of the problem space, the complexity of the requirements model progressively increased. New components and their complex relationships are elaborated and added into the model. Therefore, the essential complexity of the requirements model increases, reflecting the increasing inherent understanding of the requirements problem by the requirements engineer. This is consistent with Guindon's description of RE as a knowledge discovery process [17].
- The more complex the model becomes, the harder it becomes to add and fit new components to the growing model. The incidental complexity grows rapidly over time. This is also consistent with what is described as the increasing entropy in software engineering literature.
- At some stage, a sudden unexpected flash of insight occurs, a new way of understanding and conceptualising the problem suddenly becomes apparent. The new understanding gained by insight, referred to as the accidental complexity (Figure 4), leads to a significant change in the problem space. The problem is reconceptualised and the model undergoes a major restructuring step. In addition, the incidental (and thus overall) complexity of the model significantly drops. This effect of reconceptualisation insight can be illustrated as the dropping lines in Figure 1.
- Note that the insight results in the gain of accidental complexity (the reconceptualisation of the problem space) rather than merely the (partial) removal the incidental complexity in the

---

[3] For interested readers, two illustrative stories can be found in [37].

model. In fact, the simplification of the model should not be understood merely in terms of the reduction of the number of components of the model. Instead, the model had a new architecture reflecting a new perception of the problem by the requirements engineer.

In addition, the requirements engineer may facilitate a *post hoc* evaluation in order to leverage the effect of insights at the crisis point by constructing a retrospective examination of the problem space when insight happens. Details can be found in Nguyen and Swatman [35].

- After reconceptualisation, the newly reconstructed model becomes the basis for further development cycles.

In summary, during this process of learning about the requirements, the requirements engineer goes through a number of conceptualisation cycles starting with building up a base conceptual architecture of the model (an abstraction of the problem), then continuously concretising this architecture (perceived abstraction) by populating and modifying it with concrete conceptual tools/elements (e.g. components, objects/classes and their associations…). At some crisis point, as a result of insight the problem is reconceptualized and viewed from another perspective. Consequently, the model undergoes a major restructure with a new base architecture (abstraction) and again is re-concretised during the next building cycle.

In addition, the restructuring is primarily insight-driven rather then being based on systematic effort. The insight leads the gain of the accidental complexity and the removal of the incidental complexity in the model. As a result, the restructured model has a new architecture representing a new (more elegant) way of perceiving and understanding the requirements problem.

2.3. Discussion and related work

The discussion above describes a new, empirically though qualitatively grounded conceptualisation of the RE process. Quantitative measurement and assessment to confirm and support the finding is needed and will be undertaken in future projects. Plans and directions for future research in this respect are described in Nguyen and Swatman [37] which also reviews the catastrophe-cycle RE model in relation to current literature in general problem solving and RE.

The catastrophe cycle RE process, the literature and “approved” views

Our understanding of the RE process appears to be:

- <u>in line</u> with the intuition of the professional community
- <u>in line</u> with the literature in general problem solving, but
- <u>in conflict</u> with the commonly accepted view in the RE literature.

Firstly, the catastrophe cycle RE process <u>resonates</u> with the RE professional. Practitioners with whom we have discussed our work uniformly confirm that the real RE process is different from the “approved” smoothly balanced, incremental process. In his article about creativity in software development, McBreen [33, p.1] points out that most design documentation “*is written as if the design was developed linearly in a clean sequential manner*”, therefore we tend to assume that the “*creation process should be linear as well*”.

Secondly, the catastrophe-cycle process is <u>in line</u> with literature on problem solving, creativity, and general science. Indeed, the catastrophe cycles within the RE process is reminiscent – at a micro level – with Kunh’s [25] concepts of paradigm shifts between periods of so called “normal science”.

In addition, the dynamics of the essential complexity in the requirements model is consistent with the description of the cognitive problem solving process offered by Gigch [14] . Gigch suggests that in problem solving, the problem is processed through a number of cognitive functions. The problem is continuously interpreted and explained and the perceived complexity of the problem increases. At the reformulation points, the problem scope is expanded, and the complexity is gained at a higher level of abstraction and logic. This is consistent with our interpretation of the evolution of the essential and accidental complexity in the requirements model.

Having analysed and discussed creative activities in mathematical fields, previous thinkers like Hadamard [19] and Poincaré [40] identified four stages of invention: preparation, incubation, illumination (insight), and the verification and expression of insight. At the first stage, the consciousness works as preparatory by exploring the problem areas and shaping directions that the unconscious may follow. Incubation is often described as the period when the problem solver moves away from the problem in hope of reaching a solution. Illumination can be viewed as a breakthrough by unconscious ideas when consciousness is weakened. The instant insight is often referred to as a sudden creative thought associated with an “Aha” or the Eureka effect which solves the problem in a simple and elegant way. This invention process, together with the of gaining of complexity described by these authors is consistent with the catastrophe cycle process that at crisis points the overall complexity continually grows while the accidental complexity is gained and the incidental complexity is reduced as a result of sudden sparked insight.

In his book creativity in general design, Lawson [27, p.112] eviews design techniques used to promote creative thoughts and states that they are based on the *"idea of shifting the designer's attention and changing the context within which he perceives the problem."*

Finally, the catastrophe cycle RE process, however, is fundamentally different the commonly accepted view in the RE literature. Not only does the textbook literature describe the RE process as smoothly evolutionary and generally cumulative, but the research literature also focuses on a generally incremental model. For example, although the often cited description of the RE process by Pohl (1994) recognises different dimensions of the process, it still reflects a generally incremental, evolutionary process.

Undoubtedly, the understanding of the creativity and insight in RE needs further examination and progress. Indeed, Maiden and Gizikis [32] criticise the lack of studies in creativity in RE, review current research into creativity, and strongly argue for the recognition and need for creativity in future RE research. In the larger context of software development, Glass [15] also promotes research into creativity.

Implications of the catastrophe-cycle model

Although the catastrophe-cycle if viewed at a very high level of abstraction might look (smooth) incremental or have similar shape, the fundamental difference is the reconceptualisation in the requirements engineer's understanding of the client's requirements. It shows that the learning of the requirements is not simply the process of building knowledge by adding more information and details, but involving cycles of building up an understanding and reconceptualisation of that understanding at crisis points. Moreover, crisis points and reconceotualisation insight happen inevitably and need to be promoted, recognised and supported.

In practice:

- Deviation from the catastrophe cycle model would signal that the managerial actions may be required. The lack of shrinkage of the complexity in the requirements model would indicate the lack of reconceptualisation insight and cognitive flexibility by the requirements engineer while the excessive frequency of shrinkage of the complexity would indicate the lack of persistence in developing a mature understanding of the requirements problem by the requirements engineer.
- By ignoring the importance of crisis points, the current RE methods and CASE tools may hinder creative ideas and reconceptualisation insight. Indeed, McBreen [33, p.1] states that linear, sequential models of software development *"drastically reduces our ability to create really great software"*.
- Although, it is not clear what triggers reconceptualisation insight, RE techniques and methods should promote cognitive flexibility and support the reconstruction of the requirements model when insight happens. New, effective approaches to monitoring and managing the RE process are clearly needed.

In education, the catastrophe cycle process strongly suggests that RE requires both insight and creativity as well as technical knowledge. This provides a new challenge in RE education: how can (and should) we train REers to work effectively in an environment, where insight and creativity are required. The next section argues that traditional learning models fail to support the learning required for RE and proposes both a new model based on cognitive flexibility and a framework for RE education to support this model.

3. Towards a RE learning model

3.1. Traditional learning

This conflict between 'approved' and actual behaviour is at the root of a major dilemma in RE education, and further exacerbates the challenge of educating REers. Introductory, tertiary level texts (which may be viewed as an embodiment of the current wisdom in the discipline [20, 8]) portray the RE process as smoothly incremental. These texts form the basis of RE education, and therefore propose, implicitly, a learning behaviour that models the accepted (as opposed to actual) behaviour of professional REers.

Accepting a smoothly incremental or evolutionary approach to the RE process equates well with traditional learning theories and models. In their simplest form these, based on behavioural theory, state that learning outcomes in a domain may be attained through the right set of instructional stimuli. Response to a stimulus is predictable and reliable – all the instructor requires is to identify the subskills to be mastered so that the intended behaviour is learned and to select the stimuli and strategy for its presentation that builds each subskill [50]. Amongst others, learning may be viewed as a progression to expertise through task analysis, strategy selection, try-out and repetition. These approaches are modelled in scientific and engineering methodologies, with their focus on process and repeatability.

However, the creativity of the RE process [28, 31, 32] is hampered by strict adherence to engineering and science methodologies. These:

- restrict the essential characteristics of the process (such as opportunism) [17, p. 733]
- assist in accidentally adding complexity through their attempts to control the RE's professional practice (Sutcliffe and Maiden [46, p.735] suggest strict adherence to method procedures may restrict natural problem-solving) and
- impose a plan at odds to the RE's cognitive planning mechanisms and hence interfering with the management of knowledge (Visser [48, p. 276] suggests in practice, a plan is followed only as it is cognitively cost-effective).

The new understanding of the RE process described above provides a challenge for RE education. As revealed in this study, RE requires insight and creativity as well as technical knowledge. However, approaches to training REers based on traditional learning models tend to focus on technical knowledge, and are based largely on notations and prescribed processes. Despite the engineering and manufacturing metaphors that drive the view that software development is a smooth transformation (of input to output), it is dominated by human cognition: software development is an exploratory and self-correcting dialogue [2].

The mismatch between 'approved' and actual behaviour is supported in the literature on expert behaviour [48, 42]. Experts don't do in practice what they say the do (eg follow a methodology) because their own plans are cognitively more cost effective and flexible allowing for creativity and opportunism. Planning is described as the management of knowledge structures, and relates to both the way learning takes place and is later exploited.

These differing perspectives have major influence on the underlying knowledge structures, skills (physical and cognitive) and techniques the RE has recourse to. Just as the creativity of this process is hampered by strict adherence to engineering and science methodologies, so too the education of its proponents is hampered by adherence to traditional learning models.

A poor fit between the characteristics of the domain and those of the learning model produce an 'incorrect' learning environment, where the learner is not directed to the important features of the domain. This is seen to impact greatly on the efficacy and efficiency of further learning [16] and is especially relevant in light of the noted inadequacy of formal education in training competent analysts/designers [42].

Attempts to deal with these issues have been made in the area of software design education, where the more traditional lecture + laboratory work + assessment tasks are augmented by either a capstone project which simulates a start to finish development environment or an industry-based placement (both typically towards the completion of the qualification). These are seen to provide opportunities for both authentic and experiential learning, with emphasis not so much on acquisition of knowledge as on increasing students' ability to perform tasks.

While accepted as valuable, this approach is flawed in several respects:

- the opportunity (project or placement) is presented as an aid to content learning rather than a substitute
- it focuses on *know-how* which will allow students to gain competence to practice within given frameworks (but not necessarily outside of them)
- students are expected to transfer skills acquired to the world of work, but without them necessarily being rooted in cognitive content and professional judgement.

(based on Savin-Baden [43])

and is seen to reflect her Model II learning environment.

Table 1. Model II: learning for professional action

knowledge	practical and performative
learning	outcome focussed acquisition of skills and knowledge for the workplace
problem scenario	focussed on real-life situations that require an effective practical solution
students	pragmatists inducted into professional cultures who can undertake practical action
facilitator	a demonstrator of skills and guide to best practice
assessment	testing of skills and competencies for the work place supported by a body of knowledge

Although providing experiential learning opportunities, learning from experience is not automatic: it requires transfer (the ability to apply something learned in one situation to another setting [23]) to be enabled. This transfer is enhanced where there is a focus on metacognitive strategies and reflection. It is this facet that is often missing from capstone projects and placements.

3.2. A need for cognitive flexibility

The characteristics of the RE process, as described in the research literature and this study, namely:

- its opportunistic behaviour
- the need for model restructuring and problem reconceptualisation to deal with intrinsic complexity
- a dependence on insight and creativity

suggest that student REers require enhanced understanding of learning processes, including reflection and critical thinking in order to model the behaviour of practitioners.

Learning theories and models that focus on cognitive flexibility and exploit metacognitive learning strategies have greater potential for RE education. Cognitive flexibility includes the ability to represent knowledge from different conceptual and case perspectives and, later, the ability to construct from these a knowledge ensemble tailored to the needs of the understanding or problem-solving at hand the same items of knowledge need to be presented and learned in a variety of different ways and for a variety of different purposes. Metacognitive strategies include the development of cognitively flexible processing skills and the acquisition of knowledge structures that can support them.

These are characteristics of constructivist learning theory, based on three broad principles:

- each person forms their own representation of knowledge
- knowledge construction occurs when an inconsistency between current knowledge and experience occurs
- knowledge construction occurs within a social context.

This knowledge construction is more effective when linked to metacognitive decision-making (that brings together both a knowledge of the external tools that are being used to reason with and, perhaps implicitly, a sense of how the use of those tools fits the constraints of both situations and one's own cognition [51]) requiring a wide range of skills and experience and a process of reflection.

The perspective that suggests that REers are not given problems, they construct [49] or discover [17] them also suggests constructivist learning theory may address the challenge of educating REers.

In addition, learning models that address *wicked* domains (of which the above characteristics, as well as others, confirms RE as an example [5]) propose that a foundation in the content needs to be balanced with elements of creativity and experience based on practice. In general, these models are based around constructivist principles and more specifically on experiential learning tradition.

3.3. A framework for RE education

The ‘incorrect’ learning environment resulting from the poor fit between the characteristics of the domain and those of the learning model may be addressed through the development of a new framework for RE education.

We suggest that this framework should

- be based on constructivist theory with a focus on strategic knowledge
- be placed within a situated experiential environment where authentic context is exploited
- provide the student with a *cognitive apprenticeship* [4] with its emphasis on modelling practice and making tacit knowledge explicit.

Knowledge construction and transfer are enhanced if there is a learning focus on strategic knowledge. This includes strategies for identifying and meeting sub-goals, procedural steps as well as metacognitive strategies for directing, monitoring and evaluating learning .

Learning beyond the initial stages may best be achieved through situational case studies with rich contextual information [11]. Focussing on the solution of authentic problems as a context for learning provides students with entry to the community of practice to which they will belong

A cognitive apprenticeship allows the teacher to move from mentor to coach to facilitator as students are empowered to think independently.

The implications for the design of teaching are two-fold:

- academic learning must be situated in the domain of the objective, the activities must match that domain
- academic teaching must address both the direct experience of the world, and the reflection on that experience that will produce the intended way of representing it [26]. This framework is based on aspects of Laurillard’s learning model, which incorporate teachback [38] and self explanation [9] as key phenomena in the learning dialogue. These exploit the value of the design explanation proposed in the study [34, 35, 37] leading to enhanced understanding of RE practice through a process of debriefing and reflection on action.

An appropriate RE educational framework requires a focus on these elements:

- knowledge construction based on experience with multiple perspectives and representations
- metacognitive decision making competence, implying both wide content knowledge and reflection
- experience, within an authentic context.

An appropriate learning environment for RE should provide a number of opportunities:

- to identify, analyse and solve a number of issues, repetitively and at different levels of complexity. This acts as preparation for professional employment
- to practise the art as well as science of RE
- to test the understanding of theory, its connection with application, and develop theoretical insight
- to deal with incompleteness and ambiguity
- to think independently and work co operatively, fostering insight into individual strengths and weaknesses.

3.4. Future work

Current work is focussed on the development, testing and subsequent evaluation of such a framework, based on a learning-centred model for educational evaluation. Through a process of

- curriculum analysis – to to describe the current curriculum, its inadequacies and insufficiencies, with particular attention to the shortfall in learning
- teaching for learning analysis - to describe the teaching/learning process likely to bring about the desired learning outcome
- specification for innovation - to describe and justify the proposed implementation and indicate how it will facilitate the desired learning process and outcome

the learning needs are being documented. The RE education framework will to be tested in a teaching curriculum, and monitored in terms of

- the learning environment – is it functional in its context and accessible/attractive to students
- the learning process – is it being influenced as intended.

Future generalisation of the framework beyond its immediate context can then be based on the robustness of the learning and its transfer (impact evaluation) and a determination of its sustainability.

4. Conclusion

The poor fit between approved and actual behaviour in RE practice is mirrored by a poor fit between learning models based on a behaviouralist legacy. In an educational environment, this leads to an incorrect learning environment.

A better understanding of actual RE behaviour, and the importance of design explanation, provided through this study and elsewhere suggests that the metaphor of learning as dialogue should be exploited and suggests that a learning framework that acknowledges the place of cognitive apprenticeship better matched actual behaviour in RE.

References

[1] Avison, D. E. (1991). Action Programmes for Teaching and Researching in Information Systems, *The Australian Computer Journal* **23**(2):66-72.

[2] Bach, J. (1999) Reframing requirements analysis, *IEEE Computer*, **32**: 120-122.

[3] Batra, D., Davis, J. G. (1992). Conceptual Data Modelling in Database Design: Similarities and Differences between Expert and Novice Designers, *International Journal Man-Machine Studies* **37**: 83-101.

[4] Berkenkotter, C. and Huckin, T. N. (1995) *Genre Knowledge in Disciplinary Communication: Cognition/Culture/Power*. Hillsdale (NJ): Lawrence Erlbaum Associates.

[5] Bubenko, J.A. (1995). Challenges in Requirements Engineering, *RE'95: Proceedings of the Second IEEE International Symposium on Requirements Engineering,* IEEE Computer Society Press, Los Alamitos, CA, 160-162.

[6] Carroll, J., Swatman, P. A. (1998). The process of deriving requirements: Learning from practice, *Proceedings of the ninth annual Australasian Conference on Information Systems*, Sydney, Australia, (29 Sep- 2 Oct), 51-63.

[7] Chaiyasut, P. and Shanks, G. (1994) Conceptual Data Modelling Process: A Study of Novice and Expert Data Modellers, *Proceedings of the First International Conference on Object-role Modelling*, Magnetic Island, Australia, (July), 310-333.

[8] Checkland, P. B. and Holwell, S. (1998) *Information, Systems and Information Systems - making sense of the field*. Chichester: Wiley.

[9] Chi, M. T. H. and Bassock, M. (1989) Learning from examples via self-explanation, in *Knowing, Learning and Instruction: essays in honour of Robert Glaser*, L. Resnick, Ed. Hillsdale (NJ): Lawrence Erlbaum Assoc, pp. 251-282.

[10] Conklin, E. J., Yakemovic, K. B. (1991). A Process-oriented Approach to Design Rationale, *Human Computer-Interaction* 6: 357-394.

[11] Dreyfus, H. L. and Dreyfus, S. E. (1986) *Mind over Machine*. New York: Free Press.

[12] Galliers, R. D. (1991). Choosing Information Systems Research Approaches, *Proceedings of the IFIP TC8/WG Working Conference*, North-Holland, 327-345.

[13] Galliers, R. D. (1992). *Information Systems Research: Issues, Methods and Practical Guidelines*, Blackwell Scientific Publications.

[14] Gigch, J. P. (2000) Metamodelling and problem solving, *Journal of Applied Systems Studies*, **1**(2),: 327-336

[15] Glass, R. L. (2001). A story about the Creativity Involved in Software Work, *IEEE Software*, September/October, 16(5): 10-12.

[16] Gobet, F. and Wood, D. (1999) Expertise, models of learning and computer-based tutoring, *Computers & Education*, **33**: 189-207.

[17] Guindon, R., (1989). The process of knowledge discovery in system design, in G.Salvendy and M.J. Smith (eds), *Designing and Using Human-Computer Interfaces and Knowledge Based Systems*, Elsevier Science Publisher, Amsterdam Netherlands, pp.727-734.

[18] Guindon, R., 1990. Designing the Design Process: Exploiting Opportunistic Thoughts, *Human-Computer Interaction*, 5:305-344.

[19] Hadamard, J. (1954). *The Psychology of Invention in the Mathematical Field*, Dover Publications, New York.

[20] Iivari, J. (1991) A paradigmatic analysis of contemporary schools of IS development, *European Journal of Information Systems*, **1**: 249-272.

[21] Jeffries, R., Turner, A. A., Polson, P. G. and Atwood, M. E. (1981) The Processes Involved in Designing Software, *in* Anderson, J. R. (ed.), *Cognitive Skills and Their Acquisition*, Lawrence Erlbaum Associates Publishers, Hillsdale New Jersey USA, chapter 8, pp. 255-283.

[22] Kant, E. and Newell, A. (1984). Problem Solving Techniques for the Design of Algorithms, *Information Processing and Management*, 20:1-2, 91-118.

[23] Kearsley, G. (2000*) Explorations in Learning & Instruction: the theory in practice database*, George Washington University, Washington (DC).

[24] Khushalani, A. J. (1997). *Modelling and Supporting Opportunistic Design Problem Solving*, PhD thesis, School of Computer Science and Software Engineering Swinburne University of Technology, Melbourne Australia.

[25] Kuhn, T. (1970). *The Structure of Scientific Revolutions*, 2nd edn, University of Chicago Press, Chicago, IL.

[26] Laurillard, D. (1993) Rethinking University Teaching: a framework for the effective use of educational technology. London: Routledge.

[27] Lawson, B. (1990). *How designers think: The design process demystified*, 2nd edn, Butterworth Architecture: London.

[28] Lubars, M., Potts, C., and Richer, C. (1993) A review of the state of the practice in requirements modeling, *Proceedings of the International Symposium on Requirements Engineering*, San Diego.

[29] Malhotra, A., Thomas, J. C., Carroll, J. M., Miller, L. A. (1980). Cognitive processes in design, *International journal Man-Machine Studies* **12**: 119-140.

[30] MacLean, A., Young, R. M., Belloti, V., Moran, T. (1991). Question, Option, and Criteria: Elements of Design Space Analysis, *Human-Computer Interaction* 6: 201-250.

[31] Maiden, N. A. M.and Sutcliffe, A. G. (1992) Exploiting reusable specifications through analogy, *Communications of the ACM*, **34**: 55-64.

[32] Maiden, N. A. M., Alexis Gizikis (2001). Where do Requirements Come from? *IEEE Software*, September/October, 16(5): 10-12.

[33] McBreen, P. (1999) Creativity in Software Development, McBreen.Consulting, URL: http://mcbreen.ab.ca/papers/CreativityPaper.html

[34] Nguyen L., Swatman P. A. and Shanks G., (1999) Using Design Explanation within Formal Object-Oriented Method, *Journal of Requirements Engineering*, Springer-Verlag London Ltd, England, 4(3): 152-164.

[35] Nguyen, L. and Swatman, P.A. (2000a) Complementary Use of *ad hoc* and *post hoc* Design Rationale for Creating and Organising Process Knowledge. *Proceedings of the Hawaii International Conference on System Sciences HICSS-33*, January 4-7, Maui, Hawaii, USA.

[36] Nguyen, L. and Swatman, P.A. (2000b) Essential and Incidental Complexity in Requirements Models, *Proceedings of the fourth IEEE International Conference on Requirements Engineering ICRE 2000*, June 19-23, Schaumburg, Illinois, USA, 130-139.

[37] Nguyen, L. and Swatman, P.A. (2001). Managing the Requirements Engineering Process, submitted to *Requirements Engineering Journal*.

[38] Pask, G. (1976) *Conversation Theory: applications in education and epistemology*, Amsterdam: Elsevier.

[39] Pohl, K. (1994) Three Dimensions of Requirements Engineering: framework and its application, Information Systems 19(3):243-258.

[40] Poincaré, H. (1952) Science and Method, Dover Publications, New York.

[41] Rittel, H. W, J. and Webber, M. M. (1973) Dilemmas in a general theory of planning, *Policy Sciences*, 4: 155-169

[42] Robillard, P. (1999). The role of knowledge in software development, *Communication of the ACM* **42**(1): 87-92.

[43] Savin-Baden, M. (2000) *Problem-based Learning in Higher Education: untold stories*. Buckingham (UK): Society for Research into Higher Education and Open University Press.

[44] Schön, D. (1996). Reflective conversation with materials, in T. Winograd (ed*.), Bringing Design to Software*, New York: ACM Press, 171-184.

[45] Simon, H. A. (1973) The structure of ill-structured problems, *Artificial Intelligence*, **4**(3):181-200

[46] Sutcliffe, A. G., Maiden, N. A. M. (1992). Analysing the novice analyst: cognitive models in software engineering, *International Journal Man-Machine Studies* 36: 719-740.

[47] Swatman, P.A. (1996). Formal Object-Oriented Method—FOOM, in H. Kilov and W. Harvey (eds), *Object-Oriented Behavioural Specification*, Kluwer Academic Publishers, Boston USA, 297-310.

[48] Visser, W. (1990) More or less following a plan during design: Opportunistic deviations in specifications, *International Journal of Man-Machine Studies* **33**: 247-278

[49] Visser, W. (1992). Designers' activities examined at three levels: organisation, strategies and problem-solving processes, *Knowledge-Based Systems* **5**(1): 92-104.

[50] Winn, W. and Snyder, D. (1996) Cognitive perspectives in psychology, in *Handbook of Research for Educational Communications and Technology*, D. H. Jonassen, Ed. New York: Simon & Schuster Macmillan, pp. 112-142.

[51] Wood, D. (1999) Editorial: representing, learning and understanding, *Computers & Education*, **33**: 83-90.