

University of Wollongong

Research Online

Faculty of Engineering and Information
Sciences - Papers: Part A

Faculty of Engineering and Information
Sciences

1-1-2012

Cognitive processes in object-oriented requirements engineering practice: analogical reasoning and mental modelling

Linda Dawson

University of Wollongong, lindad@uow.edu.au

Follow this and additional works at: <https://ro.uow.edu.au/eispapers>



Part of the [Engineering Commons](#), and the [Science and Technology Studies Commons](#)

Recommended Citation

Dawson, Linda, "Cognitive processes in object-oriented requirements engineering practice: analogical reasoning and mental modelling" (2012). *Faculty of Engineering and Information Sciences - Papers: Part A*. 295.

<https://ro.uow.edu.au/eispapers/295>

Research Online is the open access institutional repository for the University of Wollongong. For further information contact the UOW Library: research-pubs@uow.edu.au

Cognitive processes in object-oriented requirements engineering practice: analogical reasoning and mental modelling

Abstract

This chapter presents a background in cognitive processes such as problem-solving and analogical reasoning for considering modelling from an object-oriented perspective within the domain of requirements engineering. This chapter then describes a research project and the findings from a set of four cases which examine professional practice from perspective of cognitive modelling for object-oriented requirements engineering. In these studies, it was found that the analysts routinely built models in their minds and refined them before committing them to paper or communicating these models to others. The studies also showed that object-oriented analysts depend on analogical reasoning where they use past experience and abstraction to address problems in requirements specification.

Keywords

practice, mental, engineering, requirements, oriented, object, processes, cognitive, reasoning, analogical, modelling

Disciplines

Engineering | Science and Technology Studies

Publication Details

Dawson, L. (2013). Cognitive processes in object-oriented requirements engineering practice: analogical reasoning and mental modelling. In R. Pooley, J. Coady, H. Linger, C. Barry, M. Lang & C. Schneider (Eds.), *Information Systems Development: Reflections, Challenges and New Directions* (pp. 115-128). United States: Springer.

Cognitive Processes in Object-Oriented Requirements Engineering Practice: Analogical Reasoning and Mental Modelling

Linda Dawson¹

Abstract

This paper presents a background in cognitive processes such as problem solving and analogical reasoning for considering modeling from an object-oriented perspective within the domain of requirements engineering. The paper then describes a research project and the findings from a set of four cases which examine professional practice from perspective of cognitive modeling for object-oriented requirements engineering. In these studies, it was found that the analysts routinely built models in their minds and refined them before committing them to paper or communicating these models to others. The studies also showed that object-oriented analysts depend on analogical reasoning where they use past experience and abstraction to address problems in requirements specification.

Keywords: Case study, object-oriented, requirements engineering, mental models, cognitive processes

1 INTRODUCTION

Information systems development (ISD) and requirements specification for ISD can be considered as socio-technical activities (Pohl 1994; Urquhart 1998; Kautz, Hansen et al. 2004). Underlying the technical and social aspects of the requirements engineering (RE) process are skills and concepts from other disciplines such as sociology, organisational science, and cognitive science (Hirschheim and Klein 1989; Checkland and Scholes 1990; Urquhart 1998). Some empirical research has addressed abductive reasoning in RE (Menzies, Easterbrook et al. 1999), and the importance of perceptions of processes by stakeholders in the RE process (Napier, Mathiassen et al. 2009). Although there have been some studies of programmers and the role of experience and knowledge in software (rather than systems) quality (Steen 2007; Gendreau and Robillard 2009), an understanding of how successful systems analysts think and produce requirements models needs to be investigated (Burton-Jones and Meso 2006; Dobing and Parsons 2006; Grant and Reza 2007).

This paper presents the findings of a multiple case study which examined cognitive modelling for object-oriented (OO) RE based on qualitative data collection and analysis methods. The data and findings presented here are part of a larger

¹ Linda Dawson, Faculty of Informatics, University of Wollongong, NSW, Australia

study of OO RE practice. In the four cases reported in this paper requirements specification involved mental modelling during the transformation from elicitation to concrete models for design and implementation. This mental modelling process involved abstraction and analogous reasoning as well as problem-solving activity.

Section 2 of this paper provides background in cognitive modelling. Section 3 provides background in OO RE. The research approach adopted is presented in Section 4. Section 5 presents the four case studies and findings and a discussion of the findings is presented in Section 6.

2 COGNITIVE PROCESSES IN THE REQUIREMENTS ENGINEERING PROCESS

Most studies of cognition and system modelling and design have been based on positivist-style surveys (Lang and Fitzgerald 2007; Thomas and Bostrom 2007; Shaft, Albert et al. 2008), or formal models (Wang and Fu 2004; Overbeek, van Bommel et al. 2007) or normative theoretical studies (Gasson 2004; Nelson and Monarchi 2007; Siau and Wang 2007).

Loucopoulos and Karakostas (1995) identified some factors affecting requirements specification as a problem solving activity as follows:

- Analysis problems are ill-defined and constantly changing as the organisational context changes and more information about user requirements is gathered
- Requirements exist in organisational contexts and may be conflicting from differing viewpoints
- The process of analysis is a cognitive activity, requiring understanding of an abstract problem, and development of a logical and internally consistent set of specifications.

Successful requirements specification can be considered to be an art or a skill honed from a great deal of experience (Loucopoulos and Karakostas 1995; Macaulay 1996; Sommerville and Sawyer 1997).

2.1 Thinking and Problem Solving

Problem modelling for simulation or systems design is an important aspect of many design activities (Schön 1983; Khushalani, Smith et al. 1994; Galal and McDonnell 1998). All of these design processes draw on the discipline of cognitive science or the study of human thinking and problem-solving. Problem modelling in systems analysis is not necessarily aimed at modelling “problems” in the general sense (although the term problem is an often-used general term as it is in cognitive science) but is usually aimed at achieving some task. That is, if there is a perception of a task to be accomplished or a problem to be solved, a systems designer must first describe or model the problem or task in a way that facilitates progress towards accomplishing that task or solving that problem.

Problem solving can be defined as having three phases (Polya 1957; Mayer 1992):

Understanding the problem in terms of what is known (givens) and what needs to be achieved (goal). i.e. describing what the task is.

Planning a solution (using past experience where appropriate). i.e. reusing previous knowledge to describe how to achieve the task.

Testing the result. i.e. validating and verifying the solution.

The investigation of thinking, problem solving and cognition has long been a major concern of cognitive psychologists, and now cognitive scientists are seeking to apply cognitive theory to information technology disciplines. General definitions of thinking and problems are a useful starting point for discussing problem modelling for information systems development.

Mayer (1992) defines human cognitive processes in terms of problems and thinking. He defines a problem as having the characteristics of givens and goals: and that “... any definition of “problem” should consist of the three ideas that (1) the problem is presently in some [given] state, but (2) it is desired that it be in another [goal] state, and (3) there is no direct, obvious way to accomplish the change.”

Mayer (1992) also considers the terms *thinking*, *problem solving* and *cognition* to be equivalent, and although there is a serious lack of agreement among psychologists about the definition of thinking, he suggests a compromise general definition that most psychologists might accept:

- Thinking is *cognitive*, but is inferred from behaviour. It occurs internally, in the mind or cognitive system, and must be inferred indirectly.
- Thinking is a *process* that involves some manipulation of a set of operations on knowledge in the cognitive system.
- Thinking is *directed* and results in behaviour that “solves” a problem or is directed towards a solution.

In other words, thinking is what happens when a person solves a problem, that is, produces behaviour that moves the individual from a given state to a goal state.

2.2 Analogical Reasoning

Analogical Reasoning is based on analogues, models and examples and “... *permeates all our thinking.*” (Polya 1957). Mayer (1992) defines analogical reasoning as abstracting a solution strategy from one problem and relating that information to a new problem where the original domain is called the *base* domain and the domain to be explained is called the *target* domain. A critical skill in the cognitive processes required in conceptual modelling is the ability to apply the principle of abstraction. Abstraction is a key aspect of analogical reasoning.

Two analogical problems may have a *surface similarity* where the two problems share common characteristics which may not be related to the solution and/or a *structural similarity* where the relations of objects in one problem correspond to the relations of objects in the other problem. If solving one problem helps to solve another problem we can say there has been an *analogical transfer* of problem

solving strategy. Mayer (1992) suggests that it is *how* the analogies are presented which influences the positive or negative outcome of analogical transfer.

Mayer describes three conditions for successful analogical transfer:

- *Recognition* - in which a problem solver identifies a potential analog (or base) from which to reason,
- *Abstraction* - in which a problem solver abstracts a general structure or principle or procedure from the base, and
- *Mapping* - in which a problem solver applies that knowledge to the target.

Mayer divides analogical reasoning into thinking using analogs, thinking using models and thinking using examples. An analogue has structural similarity but not surface similarity with a target problem.

Knowing a solution plan for an analogous problem is not very useful unless you realise that the problem is analogous to the one you are working on. Studies in this area (Gick and Holyoak 1980; Gentner 1983; Holyoak and Koh 1987; Gentner 1989) show that experience and structural similarity of problems are critical in analogical transfer.

Encouraging subjects to abstract solutions from analogs has been the subject of several studies. Gick and Holyoak (1980) propose a *schema induction theory* which suggests that it is easier to induce a general schema from experiences with structurally similar problems in different domains than from a single problem.

Mapping knowledge from a base analogy to a target analogy depends on the overall analogy transfer, that is, recognition of an analogous problem, then the abstraction of useful information that can be used to solve another problem.

2.3 Cognitive Modelling

Mayer describes a **model** of a system as a system “[which] includes the essential parts of the system as well as the cause-and-effect relations between a change in status of one part and a change in status of another part”. He uses Gentner's (1989; Gentner, 1983) *structure-mapping theory* where knowledge about one system (the base) is used to reason about another system (the target). Gentner's system consists of *objects* with *attributes* and *relations* between objects. Studies based on using a water-flow model and a moving-crowd model to understand and reason about an electrical system showed that subjects could reason about electrical circuits based on knowledge of a water-flow model or a moving-crowd model. Mayer and Gallini (1990) also showed in a series of experiments concerning reasoning about models for explaining how radar and pumping systems work, that adding pictorial models to textual descriptions improved problem solving performance by an average of over 60 percent.

The term **mental model** is defined in cognitive psychology (Craik 1943; Johnson-Laird 1983; Norman 1983) as either analogical representation or a combination of analogical and propositional representations. Preece (1994) suggests that “A *mental model* represents the relative position of a set of objects in an analogical manner that parallels the structure of the state of objects in the world.”. In the field of

human computer interaction (HCI) a mental model has been defined as “ ... *the model people have of themselves, others, the environment, and the things with which they interact. People form mental models through experience, training and instruction.*” ((Norman 1988), p17). Mental models are used extensively in HCI to explain dynamic aspects of cognitive activity and it is suggested that people build mental models of the world in order to make predictions about an external event before carrying out an action (Preece, 1994).

3 Object-oriented Requirements Engineering

OO models and approaches (Budd 1997; Henderson-Sellers 1997; Jacobson, Booch et al. 1999; Booch, Maksimchuk et al. 2007) are claimed to provide a more natural way of specifying, designing and implementing information systems based on features which include:

- a consistent underlying representation of an identified object throughout the development process
- the encapsulation of static or descriptive characteristics together with the dynamic or behavioural characteristics of an object
- the ability to model complex systems by reusing objects and object components from previously designed systems
- the incorporation of high-level data abstraction facilities including inheritance and polymorphism

In this study the interest is in how experienced analysts and developers actually use objects in requirements specification.

4 RESEARCH APPROACH

The objective of this project was to investigate whether the cognitive processes described in Section 2 above, in particular analogical reasoning and mental models are used in OO RE and if so, how these processes are used during the requirements definition process. The research questions for this project were:

- How do analysts use **analogical reasoning and analogical transfer** when producing requirements for a systems development project?
- How do analysts develop **mental models** when producing requirements for a systems development project?

The research approach was based on cases involving taped semi-structured interviews with individual requirements engineers. Each participant was interviewed several times, providing descriptive empirical data.

5 THE CASE STUDIES

The following sections describe the four cases with particular reference to the system development experience of the analyst, the development philosophy of the consultant/analyst and the evidence of mental modelling.

Participants were recruited through industry. Some participants provided contacts for subsequent participants. There has been no attempt to select participants based on specific background characteristics. The common factor is that all the participants were currently working in the field of OO requirements specification. The contextual information for each consultant interviewed is summarised in Table 1.

5.1 Findings

The presentation of findings is based on illustrated narrative style, or an oral narrative told in the first person, as described by Miles and Huberman (1994) and Myers (2010) and as used in Fitzgerald (1997) and Urquhart (1998). This approach is described as (Miles and Huberman, 1994) “...each part of the sequence is followed by a series of illustrative excerpts [quotes from the transcripts]” which does not resort to explicit coding but looks for “... key words, themes, and sequences to find the most characteristic accounts.”. Where transcript data is quoted directly the researcher's questions or interactions are shown as bold italic and the participant's as plain italic.

Case	Title	Years in RE	Client	Project
1	Senior Consultant	12 yrs	Telecommunications	Fault Mgt System
2	Consultant	14 yrs	Software developer	Life Insurance
3	Technical Manager	12 yrs	Software developer	Stock-broking
4	Principal Consultant	15 yrs	State Govt	Web based transactions

Table 1 Background Information for each consultant

5.1.1 Case 1:

The consultant considers himself to be a very experienced developer. When asked to comment on the main advantages of an OO approach to system development he replied: “*The ability to evolve code in isolation behind interfaces just seems to be the key benefit ... the ability to make significant changes to significant amounts of code and not destabilise parts of the system which you are not directly working in.*”

The client was a large telecommunications organisation. The project was a fault management system for managing planned and unplanned outages in a transmission network. It was a five-year project and had involved two and a half years of development work for the analyst. The first stage deliverables were a suite of requirements and analysis specifications. The requirements model was a use case model and there was also a prototype.

Analogy transfer and Abstraction

The methodology used for system development was an in-house OO method. It was based on other methodologies that members of the team were familiar with.

The development process was influenced by the people who were available - and the fact that they had come with considerable industry experience in this kind of system development. *“We sampled from methodologies that we were familiar with. We used bits of other methodologies as appropriate.... five of the developers had significant experience of building similar systems elsewhere... What that meant was there were three or four people who were able to contribute to a methodology that picked up bits and pieces from a number of influences... They just all brought their biases and their interests and thoughts.”*

Mental modelling

The specifications were drawn up after the consulting analyst joined the team and a user group was part of that process.

“I would say that [the class model] was drawn up quickly within the space of probably a week to two weeks. But I would say that there were fragments of that model getting developed in a couple of people’s heads for probably three months beforehand. The development of that model was not done publicly, or the first cut of it, so after that it was tossed to the team and it just diverged ... As the requirements were being collected as the requirements modelling was being done. But I would say that it was being done largely privately and it was not written down until the last minute when it was just a dump.”

Further questioning on mental modelling followed: “How and when do these mental models start forming inside your mind. You have said that this is what happens and at some point it gets turned into hard copy ... do you think in general people doing this kind of work are mulling around mental models in their head?”

“Firstly, when we talked about collecting requirements and putting them into the requirements model, we talked about thinking of objects, and ... there will be someone whose responsibility it will be ... to start casting bits of models together. So as they go through they will be listening to discussions and working on the requirements model and they may or may not be writing things down, ... You will be listening very carefully and collecting and cataloguing constraints and refining the abstractions in your mind.”

In this case the information that was used to build the mental models came from a group of users, from project meetings and the general requirements gathering process and activities. The more technical aspects, were done later on with discussion with other members of the analysis team. The consultant explicitly saw this activity as abstraction:

“You will come up with will be an abstraction in your own mind, which you probably cannot fully express, ... you wouldn’t want to. For me anyway it’s a mistake to try and rush in and write that down and stick it into a case model. ... So you might carry round an event or an account or a customer object or something, you carry around a picture of how that is shaping in your mind. Someone in a meeting or a discussion will say, ‘Of course, you know we only ever had one of these, and that will change’ and you can say Ah! Test that against my understanding of what a customer, or event or a facility or whatever the abstraction is going to be and that might either verify or it might contradict it. If it verifies it you

probably let things go and move on to the next point. If it contradicts it, you need to pick it up and mine that and get to the bottom of that.”

5.1.2 Case 2

The **consultant** is a director and a partner of a small organisation whose members work as consultants in the OO field. He had been with the organisation for just on a year and had been doing RE for about 22 years. The consultant did not use any specific methodology in this project or any other projects. *“I haven’t been, let’s say, an advocate of any particular methodology from start to finish ... See I don’t believe in methods as such ... What I talk about is a underlying concept rather than a methodology ... you need to have rigour and the diagramming notations and the steps in the methodologies give you that, but you also need to play in the sandpit”*.

The client was a commercial software developer that builds generic packages and then sells them on to clients in the financial sector. The organisation believed that it needed to move to OO systems. The consultant was engaged *“ ... not so much as a requirements analyst but more as an OO mentor. Now in that role one of the first things we did was requirements and then moved on to design.”* The objective of the project was to produce a receipting component for a larger insurance system that would be a commercially saleable stand-alone product.

Analogy transfer and Abstraction

A discussion about how the consultant went about initial modelling of requirements led to the following illustrative comment about the usefulness and necessity of ad hoc or informal models *“ ... and in every project I’ve ever worked on be it a mathematical project or software development project there’s been a few key pictures. The one I’m working on at the moment is the billing cycle – it’s a wheel and its got the steps in the billing cycle on it and that’s in everybody’s head and everybody talks in those terms and it’s just the key base thing – it’s the conceptual core of the thing ... I’m a great believer in ad hoc diagrams that give the picture that springs from your understanding of the problem and in a lot of OO work the process of development hinges on one or two of these pictures. ... [and] the trouble with that [using ad hoc diagrams] as a methodology is that its difficulty is that you can’t capture it, you can’t describe it in some way that anyone [else] can really use it and that’s precisely its strength because it handles those parts of the things that don’t fit in the normal descriptions and every project’s got an aspect like that.”*

Mental modelling

The consultant saw OO modelling as *“ ... a superb way of modelling the real world”* which allows a high level of abstraction *“ ... I’m unlearning some of my [data modelling] prejudices being back with [my old colleagues] and starting to look at objects and classes more in terms of services than as data and deferring the internal structure later and later and later into the design.”*

“I think that I do immediately start thinking of key objects during requirements gathering, not in any formal way, they just pop into one’s head.”

5.1.3 Case 3

The **consultant** had been involved in OO systems for about five years. He believed that although OO approaches have certain advantages they also have some limitations for developing systems *“The problem with using DFDs is that the models were far too data-centric which was fine if you were doing a lot of retrieval but to do good transaction processing was quite awkward and you really did need very high levels of expertise to get it right in the RDBMS world. That is much less so [in OO] and therefore you can end up with much simpler solutions with OO techniques as long as you keep the ‘propeller heads’ so to speak away, you actually end up with systems which are very easy to understand and easy to maintain.”*

The **client** project was a back office system for stock-brokers configurable to individual client needs. It was neither an off the shelf package nor a one-off so there was a set of core requirements. Assistance was provided by the organisation to clients in customisation and ongoing support.

Analogy transfer and Abstraction

The consultant believes that as a professional much of his requirements technique comes from his knowledge and experience on other projects. *“It works in two ways. There is using your experience to recognise ‘Hang on I’ve seen something like that before ... Yes, I recognise what it’s doing therefore I can do it’. Secondly I know ... I don’t have the same view of the business as a user because I’m used to the view from the software development side ... the side of the business I understand and therefore when I’m requirements gathering I actually understand what the user is talking about and can actually relate that to software development so therefore I can make sure I try and work my requirements gathering around ‘OK, if I had to build that what information would I need? Have they given me enough? If I built that what are the exception cases? There are some of the questions about what they want to do if it doesn’t meet these criteria, all of that. And that’s predominantly drawn on a nearly 20 year career and 12 years in the same environment.”*

Mental modelling

It was not until the requirements gathering was finished that the consultant or team started UML modelling *“... all through requirements gathering we are talking textual, primarily.”* When questioned about building mental models at this elicitation stage this consultant agreed *“ ... I don’t put anything on paper – that these mental models are sort of living in the back of the mind and as new information comes in that it sort of alters or adds to that mental model... and its very abstract and not really anything that is put on paper or shown anyone.”*

5.1.4 Case 4:

The **consultant** worked for a business consultancy which provides IT consultancy and educational services to a broad range of clients. The organisation's philosophy is outlined on their web site as: *“We do not subscribe to a single, rigid methodol-*

ogy. *Each assignment is treated as a unique challenge. We tailor our approach to meet the specific requirements of each client, drawing on a wide range of well-researched techniques and the combined experience of our consultants.*"

The **client** was a central support group within government which was assisting other departments or client organisations with the implementation of on-line service delivery to the general public. The client organisations were independent organisations within the government that provide or sell products or services to the general public.

Analogy transfer and Abstraction

In this project the analogy transfer and abstraction was explicitly built in to the RE process. The client organisations were given a general pattern for a transaction that could be configured to how the client/end client wanted it. *"There is one general methodology, one [generic] object model and there are a set of seven different templates, for each [common] transaction type that you can use and you can tailor the templates ... i.e. they are half filled out - it is not a blank form."*

The first task for the client was to work through the general use case flow diagram and instructions to see how well their transaction matched the common model. This "goodness of fit" test was important because pricing was based on it. The more variation from the common model the more it cost the client organisation.

Customising the template involved modifying the basic flow diagram (based on the "goodness of fit") and the object model, modifying the use case script by striking out (not removing) elements, so that someone could look across the page and see what had been changed. Few modifications were made to the object model by client organisations. The generic model seemed to apply to most situations. The generic templates presented an abstract view of transactions which could be mapped to different end client systems as necessary.

Mental modelling

The elicitation process was not seen as specifically OO but this consultant claimed to think in terms of objects at this stage because *"...that's the way I think ... so it's hard for me to unbundle it... we don't say to them [the client] we are really talking about objects and we are using an OO methodology. We just do it and they just want to specify their transactions"*.

6 DISCUSSION

The findings from this study show that modelling for OO requirements specification relies on cognitive skills including abstraction and mental modelling together with problem-solving and reasoning skills particularly analogical reasoning skills on the part of the analyst.

In the four cases reported in this paper requirements specification involved mental modelling during the transformation from elicitation to concrete models for design and implementation. Overall the four analysts believed that they were continually "modelling in the mind" during the elicitation process and that these mental mod-

els were further refined in the mind before they were communicated to others (users or fellow analysis team members) or before they were committed to paper. The consultants examined and discussed problems encountered with the users during elicitation and then modelled the solutions to those problems in their own minds before committing to concrete models used for design and implementation. This mental modelling process appeared to involve abstraction and analogous reasoning as well as problem-solving activity.

This study highlights several factors which are important in understanding the practice of OO RE in particular and RE in general:

Analogical Transfer: a critical skill in the cognitive processes required in conceptual modelling and conceptual reuse is the ability to apply the principle of abstraction where a problem is seen as a general instance based on a basic known analog. The recognition of structural similarity and analogical transfer may be a factor for experienced analysts when approaching new problems or systems development projects. Gentner's (1983) structured mapping theory which consists of objects with attributes and relations between objects can be mapped directly to OO modelling approaches for RE.

Cognitive mental modelling: Mental modelling used by analysts when developing requirements or design models where models are tested privately before developing more concrete paper-based models is consistent with general cognitive modelling theory in both problem solving theory and cognitive psychology.

As with many professional activities involving analysis and design (Schön, 1983, Khushalani et al., 1994, Johnston, 1999, Galal, 1998), OO RE and RE in general can be considered to be a creative process (Lawson 2006) particularly on the part of the requirements engineer or the analyst undertaking the requirements specification. Creative modelling as demonstrated by the analysts in this study relied on cognitive skills including: abstraction and mental modelling and; problem-solving and reasoning skills particularly analogical reasoning skills on the part of the analyst.

Also, the RE process is fundamentally a social process involving two main groups: the users/clients and the professional consultants (Urquhart, 1998, Loucopoulos and Karakostas, 1995, Macaulay, 1996) and this social process requires understanding by all parties to reach agreement. The facilitation of understanding and agreement requires creative modelling skills on the part on the analyst to produce understandable usable models.

It is clear that the consultants in this study demonstrated Mayer's three conditions for successful analogical transfer:

- *Recognition* – recognising similarities with previous problems or system characteristics
- *Abstraction* – abstracting a solution from one or more previous systems, and
- *Mapping* – using the characteristics of this solution in the new system problem space.

Understanding this analogical transfer in RE may allow system designers to develop appropriate tools and modify to explicitly use this process in future or modified development methodologies.

7 CONCLUSION

The research project described in this paper has provided evidence that mental modelling based on abstraction and analogous reasoning is used by professional analysts in the development of requirements specifications for system development. Although this study was specific to OO system development it has implications for requirements engineering in general. It has shown that requirements engineering like other professional analysis and design activities is a highly cognitive and creative activity.

A longitudinal study of professional systems developers – using both OO approaches and non-OO approaches, is currently underway. An investigation of emerging techniques and approaches to requirements specification and system development such as agile development and extreme programming are being incorporated in this study. The long term aim is to gain more knowledge of how large real world organisational systems are being specified and developed and which tools and techniques facilitate the successful specification for the design and implementation of these systems.

8 REFERENCES

- Booch, G., R. Maksimchuk, et al. (2007). Object-Oriented Analysis and Design with Applications.
- Budd, T. (1997). An Introduction to Object-Oriented Programming. Reading MA, Addison-Wesley.
- Burton-Jones, A. and P. N. Meso (2006). "Conceptualizing systems for understanding: An empirical test of decomposition principles in object-oriented analysis." Information Systems Research 17(1): 38-60.
- Checkland, P. and J. Scholes (1990). Soft Systems Methodology in Practice. Chichester, Wiley.
- Craik, K. J. W. (1943). The Nature of Explanation. Cambridge, Cambridge University Press.
- Dobing, B. and J. Parsons (2006). "How UML is Used." Communications of the ACM 49(5): 109-113.
- Fitzgerald, B. (1997). "The use of systems development methodologies in practice: a field study." Information Systems Journal 7: 201-212.
- Galal, G. and J. T. McDonnell (1998). A Qualitative View of Requirements Engineering. Third Australian Conference on Requirements Engineering, Geelong, Australia.
- Gasson, S. (2004). A Framework For Behavioral Studies of Social Cognition In Information Systems.

- Gendreau, O. and P. N. Robillard (2009). "Exploring Knowledge Flow in Software Project Development." International Conference on Information, Process, and Knowledge Management: Eknow 2009, Proceedings: 99-104.
- Gentner, D. (1983). "Structure mapping: A theoretical framework." Cognitive Science 7: 155-170.
- Gentner, D. (1989). The mechanisms of analogical learning. Similarity and analogical reasoning. S. Vosniadou and A. Ortony. Cambridge, England, Cambridge University Press.
- Gick, M. L. and K. J. Holyoak (1980). "Analogical problem solving." Cognitive Psychology 12: 306-355.
- Grant, E. S. and H. Reza (2007). Towards the development of a rigorous model-driven domain-specific software engineering environment. Third IASTED International Conference Advances in Computer Science and Technology, Phuket, Thailand.
- Henderson-Sellers, B. (1997). A Book of Object-Oriented Knowledge. Upper Saddle River, NJ, Prentice-Hall.
- Hirschheim, R. A. and H. K. Klein (1989). "Four paradigms of Information Systems Development." Communications of the ACM 32: 1199-1216.
- Holyoak, K. J. and K. Koh (1987). "Surface and Structural Similarity in Analogical Transfer." Memory and Cognition 15: 337-340.
- Jacobson, I., G. Booch, et al. (1999). The Unified Software Development Process. Reading, MA, Addison Wesley Longman.
- Johnson-Laird, P. N. (1983). Mental Models. Cambridge, Cambridge University Press.
- Kautz, K., B. Hansen, et al. (2004). "The Utilization of Information Systems Development Methodologies in Practice." Journal of Information Technology Cases and Applications 6(4): 1-20.
- Khushalani, A., R. Smith, et al. (1994). "What happens when designers don't play by the rules: Towards a model of opportunistic behaviour and design." Australian Journal of Information Systems 1(2): 2-31.
- Lang, M. and B. Fitzgerald (2007). "Web-based systems design: a study of contemporary practices and an explanatory framework based on "method-in-action". " Requirements Engineering 12(4): 203-220.
- Lawson, B. (2006). How designers think : the design process demystified / Bryan Lawson. Oxford ; Boston ;, Architectural Press.
- Loucopoulos, P. and V. Karakostas (1995). Systems Requirements Engineering. London, UK, McGraw-Hill.
- Macaulay, L. (1996). Requirements Engineering. London, Springer-Verlag.
- Mayer, R. E. (1992). Thinking, Problem Solving, Cognition. New York, W.H. Freeman and Company.
- Mayer, R. E. and J. Gallini (1990). "When is an illustration worth a thousand words?" Journal of Educational Psychology 82: 715-726.
- Menzies, T., S. Easterbrook, et al. (1999). "An empirical investigation of multiple viewpoint reasoning in requirements engineering." Ieee International Symposium on Requirements Engineering, Proceedings: 100-109.
- Miles, M. B. and A. M. Huberman (1994). Qualitative Data Analysis: An Expanded Sourcebook. Thousand Oaks, CA, Sage Publications Inc.
- Myers, M. (2010) "Qualitative Research in Information Systems." 1999.

- Napier, N. P., L. Mathiassen, et al. (2009). "Combining Perceptions and Prescriptions in Requirements Engineering Process Assessment: An Industrial Case Study." IEEE Transactions on Software Engineering **35**(5): 593-606.
- Nelson, H. J. and D. E. Monarchi (2007). "Ensuring the quality of conceptual representations." Software Quality Journal **15**(2): 213-233.
- Norman, D. A. (1983). Some observations on mental models. Mental Models. D. Gentner and A. L. Stevens. Hillsdales, NJ, Lawrence Erlbaum Associates: 7-14.
- Norman, D. A. (1988). The Psychology of Everyday Things. New York, Basic Books.
- Overbeek, S., P. van Bommel, et al. (2007). "Characterizing Knowledge Intensive Tasks indicating Cognitive Requirements; Scenarios in Methods for Specific Tasks." Situational Method Engineering: Fundamentals and Experiences **244**: 100-114.
- Pohl, K. (1994). "The Three Dimensions of Requirements Engineering: A framework and its applications." Information Systems **19**(3): 243-258.
- Polya, G. (1957). How to solve it. Garden City, New York, Doubleday/Anchor.
- Preece, J. (1994). Human-Computer Interaction. Wokingham, UK, Addison-Wesley.
- Schön, D. A. (1983). The Reflective Practitioner: How Professionals Think in Action. New York, Basic Books.
- Shaft, T. M., L. J. Albert, et al. (2008). "Managing change in an information systems development organization: understanding developer transitions from a structured to an object-oriented development environment1." Information Systems Journal **18**(3): 275-297.
- Siau, K. and Y. Wang (2007). "Cognitive evaluation of information modeling methods." Information and Software Technology **49**(5): 455-474.
- Sommerville, I. and P. Sawyer (1997). Requirements Engineering: A good practice guide. Chichester, John Wiley and Sons.
- Steen, O. (2007). "Practical knowledge and its importance for software product quality." Information and Software Technology **49**(6): 625-636.
- Thomas, D. M. and R. P. Bostrom (2007). The Role of a Shared Mental Model of Collaboration Technology in Facilitating Knowledge Work in Virtual Teams. 40th Annual Hawaii International Conference on System Sciences (HICSS'07). Hawaii: 37.
- Urquhart, C. (1998). Analysts and Clients in Conversation: Cases in Early Requirements Gathering. Nineteenth International Conference on Information Systems, Helsinki, Finland.
- Wang, X. C. and X. L. Fu (2004). "The cognitive skill acquisition of information reduction." International Journal of Psychology **39**(5-6): 57-57.