Applying Psychological Principles to Support Novice Conceptual Modelers

Simon McGinnes
School of Computer Science and Statistics, Trinity College Dublin
Phone: +353 1 896 2092 Fax: +353 1 677 2204
Email: Simon.McGinnes@cs.tcd.ie

Abstract

Conceptual modeling is a key skill for the designers of business information systems; conceptual modeling techniques include UML class diagrams, entity-relationship (E-R) diagrams and object role model (ORM) diagrams. It is usually easy to perform conceptual modeling on simple problems but it becomes much more difficult in real, non-trivial business situations; training and experience are required. Increasingly, however, conceptual modeling is being performed by non-experts—including those without any formal IT skills, due in part to continuing growth in software development worldwide and also to the use of desktop data management products such as Microsoft Access.

In this study, ideas from psychology were applied with the goal of making it easier for non-expert modelers to produce complete and correct conceptual models. A design framework consisting of twenty-nine psychological principles was formulated, following a review of the literature on cognition and group dynamics. The design framework was used to formulate and justify the design of an experimental modeling technique, Business Concept Modeling (BCM), and a supporting software tool. BCM incorporates the innovations of innate concept types and predictive modeling; it is similar in scope to object modeling but differs in manner of representation and method of application.

BCM and the software tool were tested alongside conventional object modeling in a series of field experiments in which an expert modeler and nine non-expert modelers used both techniques in real business situations. Qualitative and quantitative data were gathered using participant observation, questionnaires and interviews, and by analyzing the resulting models and their evolution. The effectiveness of modelers was compared between the two modeling techniques. The results show that, while the expert modeler produced models of excellent quality using both techniques, non-expert modelers were able to produce good quality models only with BCM; the models they created using object modeling were so poor as to be effectively unusable. Productivity was universally greater when using BCM, by approximately 150% for the expert and 450% for non-experts.

The results indicate that we can substantially improve the usability conceptual modeling techniques and the effectiveness of modelers using them. It appears that conventional modeling techniques such as UML and E-R are essentially too difficult to be used effectively—except by those with significant training and expertise. Yet this study also suggests that conceptual modeling itself is not inherently difficult; these techniques need not require great expertise if they can be adapted according to psychological principles. Future tools and techniques based on these principles could help non-experts to create non-trivial information systems with less need for skilled IT specialists. Such tools would not necessarily replace techniques such as UML and E-R but could offer a more usable "front end" for less skilled practitioners to apply.

Keywords

Information systems; conceptual modeling; design research; design science; requirements analysis; ontologies; systems analysis; software design; database design.

1. Introduction

1.1. Motivation for This Research

In the world of information systems (IS) development, conceptual modeling has long been used to help analysts and users to develop a shared understanding of business domains, and to design software such as databases, ontologies, middleware and user interfaces. Modeling allows information about an organization and its business concepts to be captured in structured form. Most modeling techniques use diagrams, which have historically been considered easier to understand: *"the chief merit in a diagrammatic technique is in user communication"* (Olle et al 1991). The importance of using highly formal and rigorous techniques has often been stressed: *"descriptions should be stated in a formalism with unambiguous syntax which can be understood by a suitable processor"* (Loucopoulos and Zicari 1992). Other opinion has held that models should be highly abstract so as to guarantee efficient solutions, with intuitive modeling being seen as error-prone and risky (Esculier and Friesen 1995).

More recently, doubt has been shed on the benefits of formality and rigor in conceptual modeling techniques. Diagrams, though correct, may be misunderstood; more formal models seem less useful for communication (Moody 2004; Hitchman 2004). It may be unclear why, or for whom, diagrams are being developed (Bell 2005). Clearly, further study is needed in this area (Topi and Ramesh 2002). The conceptual modeling techniques we use today have developed over many years of practice in a rapidly-evolving software industry. Perhaps now is a good time to look again at conceptual modeling, questioning some basic assumptions and taking a fresh perspective with an appropriate theoretical and empirical basis (Remenyi and Williams 1996).

1.2. Conceptual Modeling (Definition)

The term *conceptual model* is used in this paper to mean any object model or data model produced in IS development, when the model exists primarily to define mental concepts as opposed to software structures. This includes ontologies and other expressions of data structure that are intended to mimic the structure of mental concepts. The mental concepts in question relate to things that an information system needs to store data *about*; they are typically defined in an abstract way as object classes or entity types that correspond to named business concepts. This definition of conceptual models encompasses both UML class diagrams and E-R modeling. It excludes techniques that are not primarily intended for definition of mental concepts, such as use cases and the "conceptual models" of Soft Systems Methodology (Wilson 2001).

1.3. Conceptual Modeling and Subjectivity

Conceptual modeling is sometimes thought of as the capture of facts about reality; this is essentially a realist or objectivist point of view. But conceptual models normally represent the business *as perceived by* one or more end users (Everman 2005). Modeling is usually responsible for the formalization of previously intuitive thought; participants develop their ideas into fully-fledged concepts during the process. This, more realistic, view of conceptual modeling represents a subjective, nominalist position (Burrell and Morgan 1979). It has been claimed that IT professionals understand *"the implicit non-objectivist issues"* in modeling but that researchers *"tend to ignore them"* (Hitchman 1997).

Psychology tells us that our experience of reality is inevitably subjective and dependent on context. We may assume that our senses give us access to reality but experience of the world is mediated by perceptual mechanisms which guarantee unconscious distortion; experiencing and interpretation are one and the same (Goldstein 2005). This insight has deep relevance to conceptual modeling. It means that every conceptual model represents a *perspective* and there is no "correct" or "actual" structure to model. Of course, that is not to deny the physical world in which people, places and physical objects exist. But the subject matter of conceptual modeling is the day-to-day business of organizations—things like business plans, agreements, job titles, organizational structures and business transactions. These things exist largely in the mind and their definition relies on consensus. Hence we refer to the *social construction of reality*; each person has their own perception of the world and organizational truths exist by agreement only (Berger and Luckman 1966). In the context of information systems this is far more than just a philosophical point. Its significance for conceptual modeling is that we must find ways to expose and capture the *mental* concepts of business end users; we cannot simply hope to observe reality and to document it in our models. The distinction between mental concepts and reality is a crucial one for conceptual modeling, yet may be one that has received insufficient attention in the IS research sphere thus far.

1.4. Why Improve Conceptual Modeling Practice?

Conceptual modeling is one of the most fundamental skills for an IS designer. But conceptual modeling is difficult. It is an expert task, often performed by trained knowledge engineers, data analysts and system architects. Modeling skills are in demand, with continued growth in the use of databases, object-oriented software and ontologies. Yet, historically, many IT practitioners have been weak conceptual modelers (Venable 1996). The UML class diagram is perhaps the conceptual modeling technique most widely-used today—yet a solid understanding of object-orientation is necessary to use it. In all popular conceptual modeling techniques, diagrams are often complex and restructuring them can be onerous. It can be difficult to apply modeling techniques with the speed and flexibility demanded by modern development methods. It has long been recognized that techniques like UML are typically off-putting for business users and novice modelers (Bansler and Bødker 1993). All of this gives us good reason to investigate ways of making models more understandable and making modeling easier to do.

The practice of conceptual modeling has been rather stable in recent decades. Most modeling methods are descended from techniques that emerged in the 1960s and 1970s in a very different technological landscape. For example, "box and arrow" notations like class diagrams are designed for pencil-and-paper or whiteboard; they were not intended to capitalize on the visual richness and interactivity that software tools can provide (McGinnes 1994). It could be that the very stability of modeling methods has led us to see conceptual modeling as inherently difficult and something that cannot be made easier. Or perhaps we are simply applying out-of-date tools; as in many spheres, once-useful business processes can threaten operational effectiveness when circumstances change. Table 1 suggests ways in which some symptoms of business process problems might apply to conceptual modeling (Hammer and Champy 2003); we feel that this analysis supports a case for rethinking the conceptual modeling task to help make it easier and more available as a skill to a wider audience.

Table 1 Conceptual Modeling as a Candidate for Process Reengineering

Symptom	Relevance to conceptual modeling
Complexity	To be practiced well, modeling requires extensive training and experience. Models use complex notations and different diagram types may be interrelated in complex ways.
Extensive information exchange	IT specialists represent a bottleneck in the process of acquiring new systems. Business information must be communicated to an analyst who translates it into one or more models, which must then be rendered back into users' terms for verification. Models are also translated into system specifications for technicians to work with.
Data redundancy	The same information is encoded multiple times in different models (e.g. the concept purchase may be represented in a data model as a data entity and in a functional model as a process). Facts may be expressed in prose for users, in diagrammatic form, and in technical form for technicians.
Rekeying	Models on whiteboard or flipchart may be transcribed to paper and/or keyed into a modeling tool. The same or related information may later be keyed into development tools.
A high ratio of checking and control to value-adding	Models must be reconciled (e.g. data models checked against process models). The analyst must ensure that users understand models sufficiently well to be able to check them against their own view of the business. Effort must be devoted to ensuring formal correctness in models, while the ultimate benefits of doing so are unproven.
Poor quality end results	The end result is highly dependent on the skill and insight of individual analysts. Novice analysts and end users typically produce poor quality models, which translate into poor software systems.

1.5. Design Science Research

This study is an example of design science in information systems research. Design science research is distinguished from design per se by the "*production of interesting … new knowledge*" (Vaishnavi and Kuechler 2006). In other words, we approach a relevant design task in a reflective manner and take the opportunity to derive useful insights from the experience. While philosophical underpinnings are important, the IS research community can also make a positive contribution using design research to produce useful "artifacts" (Orlikowski and Iacono 2001). In this study we have generated several artifacts of interest including a design framework, containing psychological principles for the construction of conceptual modeling techniques, and a modeling technique based on the design principles. Our research approach is summarized in Table 2, which is based on guidelines for design science research in information systems (Hevner et al 2004). We see the present study as part of an ongoing, iterative research process in which the results will be recycled to formally produce and test a revised set of principles and a revised modeling technique.

Table 2 Hevner et al's Guidelines for Design Science Research

Guideline	This Study Evaluated
Design as an Artifact	This study entailed the design of three artifacts: 1. Design framework for conceptual modeling techniques (containing psychological principles) 2. BCM conceptual modeling technique & supporting tool 3. Metrics and a method of evaluating model evolution.
Problem Relevance	Conceptual modeling is difficult, yet there have been few innovations in conceptual modeling in recent years. Increasing numbers of non-experts are designing systems. Trained IT staff are in limited supply and represent a bottleneck in the system acquisition process. There is a pressing need for modeling techniques that can be used by non-experts, including those who use such techniques only implicitly as part

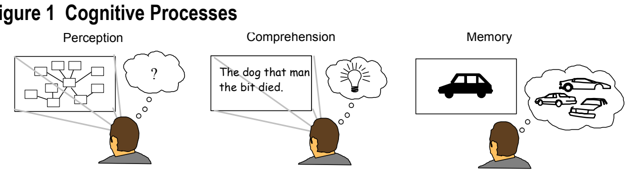
	of some other task. There is also a need for research that looks more closely into the problems of conceptual modeling in practice.
Design Evaluation	The three major artifacts were subject to evaluation in this study: 1. The design framework was evaluated both through argumentation using psychological theories and by being used to formulate the BCM technique. 2. The BCM technique & tool were evaluated in field experiments using both quantitative and qualitative data gathering. 3. The metrics and evaluation method were evaluated through use.
Research Contributions	The overarching contribution of this study is to show how our current conceptual modeling techniques are far from optimal and how they can be made available as a skill to a wider audience. We also present the psychological principles, the BCM technique, the metrics and model evaluation method and the "three patterns" analysis, all of which may be useful in subsequent research and/or practice.
Research Rigor	Rigor was applied in the selection of psychological principles through a systematic, comprehensive approach and the use of "bracketing". In the development of the BCM technique rigor took the form of a conscientious application of the chosen design framework. In the experimental portion of the study, rigor was introduced through bracketing and careful experimental design and measurement including due attention to sources of experimental bias, validity and reliability.
Design as a Search Process	This study could be described an example of satisficing in practice (Simon 1996). We have sought a solution to the "non-expert conceptual modeling" problem. This search involved the selection of suitable psychological principles, the synthesis of a modeling technique according to the principles, and the search for appropriate metrics to measure the results. In each case, alternatives were evaluated.
Communication of Research	We hope that this paper is organized such that the main messages are communicated suitably for both a managerial and technical audience. Following Zmud (1997) and Hevner (2004) we have included technical details in appendices, allowing the main elements and especially the conclusions of this research to stand out more clearly in the main body of the paper.

1.6. Relevance of Psychology

To improve the practice of conceptual modeling we must understand it—not just in theory but as a working process. In other words it is insufficient to talk about how people *ought* to model; we must study how they actually model in practice. This implies, amongst other things, that we need to understand what happens in each participant's mind as models are developed and used (remembering that many models are produced in or by groups). One major goal of conceptual modeling is communication. To know how well communication is achieved we must consider how models are perceived and understood by individuals and groups.

These are all areas that psychology can address. Psychology can help explain our cognitive strengths, such as automatic visual recognition, as well as our many cognitive limitations, such as constraints on attention and short-term memory (Solso, MacLin and MacLin 2004). Psychology illuminates the relationship between business knowledge and mental models (Johnson-Laird 2005); it helps us understand the cognitive demands of modeling and helps explain why modeling is difficult for non-experts (Figure 1).

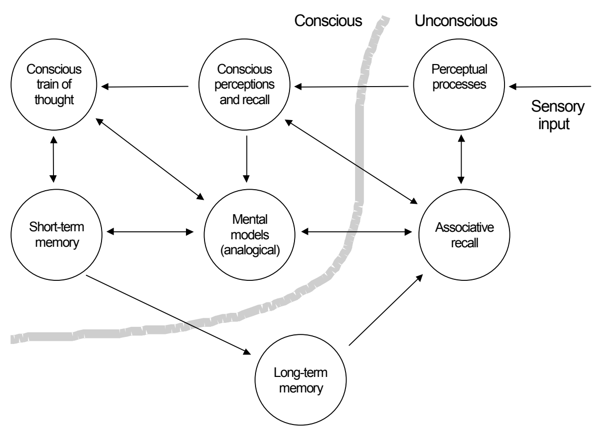
Figure 1 Cognitive Processes



Psychology is already an important reference discipline for information systems. But for conceptual modeling in particular, psychology is pivotal. "Real-world modeling practices can be informed by a deep understanding of cognitive facts" (Ramesh, Parsons and Browne 1999; Veres and Mansson 2005).

The comprehension of a model is primarily an unconscious process; it requires that perceptions be associated to memories, as illustrated in Figure 2. When we view a model, words and pictures stimulate associative recall from memory, creating conscious meaning (Solso, MacLin and MacLin 2004). This process relies on the presence of understandable words and pictures. It means that, depending on the representation used, interpreting a model can be analytical (difficult) or automatic (easy). *"Decisions regarding the presentation of conceptual models are far from trivial and should be approached with as much care as decisions on their content"* (Moody 2004).

Figure 2 Interaction of Cognitive Processes during Perception



2. Theory Development and Testing

2.1. Design Framework (Psychological Principles)

Following a review of the literature on cognition and group dynamics, we formulated a design framework consisting of psychological principles with relevance to conceptual modeling. Space does not permit a full account of this review but a representative selection of the resulting principles is outlined in Appendix A and summarized in Table 3. The principles represent distinct but related aspects of psychology as it applies to conceptual modeling; hence several themes recur in different guises.

Table 3 Selected Psychological Principles

Principle	Brief explanation
Preattentive Processing	**Maximize bandwidth using visuals** Capitalize on automatic (parallel, unconscious) cognitive processing such as visual pattern recognition and avoid the need for analytical (sequential, conscious) thought where possible.
Isomorphism	**Reduce cognitive load by matching conceptual and mental models** Avoid the need for mental translation; use constructs which match concepts that participants are likely to have already (typically, their everyday business concepts). Avoid systems-related concepts such as "object", "table" etc.
Reinforcement	**Maximize comprehension by combining words with pictures** Synergistically increase the ease and speed of recognition—and therefore comprehension—by linking text with corresponding, recognizable images wherever either images or text can appear in a model representation.
Consistency	**Consistently use each symbol with only one business meaning** Allow a mental association between a symbol and its meaning to form through repetition. Avoid confusing this association by using multiple symbols per mental concept or multiple concepts per symbol.
Chunking	**Support short-term memory with visual chunking strategies** Avoid overloading limited attention resources; present the available items of information at any point as a small number of groups, where each group corresponds to a unique mental concept and has a recognizable label and/or symbol. Allow alternative grouping methods.
Fuzzy Categories	**Support alternative concept definitions and concept instability** Allow concepts to be defined in a variety of ways such as enumeration (extensional definition), statement of rules (intentional definition) and exemplar objects, and any combination of these. Allow concepts to be redefined at will without penalty.
Brainstorming	**Capture unstructured ideas and allow easy model restructuring** Allow informal concept definitions, notes, pictures, annotations, links, etc. to form part of the formal model and to be transformed and transferred freely. Allow alternative structures to be built up and torn down easily and quickly at low cost.
Error Tolerance	**Tolerate and reduce the likelihood of simple errors** Offer a minimal construct set to avoid ambiguity about representation; prevent redundancy by eliminating ways of representing the same information more than once. Highlight (and avoid penalties for) internal incorrectness, incompleteness and inconsistency. Use automated reasoning to default structures correctly. Provide modeling guidelines and restructuring suggestions.
Arousal and Attention	**Use all means available to maintain attention** Maintain arousal at a moderate level; introduce color, sound, texture and movement (e.g. animation) in the user interface. Increase personal relevance by giving users control over modeling and image selection. Allow variation and enforced or elective interruption in task structure.
Set	To instill helpful sets, make the purpose and method of modeling obvious Specify models entirely in the user's own business terms and language. Avoid IT jargon such as "object", "entity", "relationship". Pick a visual metaphor for model representation and manipulation that the user will understand without explanation. Give early feedback, including end results (e.g. prototype user interface or database structures).

In summary the principles cover concerns such as the use of preattentive processing (automatic, parallel and unconscious mental processing); improving the match between conceptual and mental models; maximizing comprehension by combining words with pictures; associating symbols and concepts more consistently than at present; supporting short-term memory with visual chunking strategies; providing alternative concept definitions and reflecting concept instability; allowing the capture of unstructured ideas and easy model restructuring; tolerating and reducing the likelihood of simple errors; maintaining arousal and attention, and instilling helpful mental sets in the minds of modeling session participants.

The process of formulating the psychological principles was exhaustive, with many candidates being considered before a final set of twenty-nine principles was chosen. Principles were adopted if they seemed likely to have some potential impact on modeling practice and no attempt was made to prioritize them. In the following sections we describe how we applied the principles to design a modified conceptual modeling technique; following this we present an account of how the technique was tested in practice and we then reflect upon the results.

2.2. Experimental Modeling Technique

To evaluate the impact of the psychological principles in practice we formulated an experimental modeling technique, called Business Concept Modeling (BCM). BCM is a simplified version of object or data modeling, with certain innovations that can be justified in terms of the psychological principles (McGinnes and Amos 2001). BCM relies on a software tool to satisfy the various principles as far as possible. The business of defining BCM was conducted through "informed trial and error"; various ideas were examined and the psychological principles were used both as inspiration and as support for argument in favor of (or against) each idea. Since the study was completed, BCM has been used on many commercial software development projects and we have continued to modify it during this time based on experience gained.

As Hevner et al (2004) rightly point out, one cannot sensibly do research about information technology without paying some attention to the technology itself. Space does not permit a full account of BCM in this paper; however, we present a summary in Appendix B. We hope to publish fuller details of BCM in due course. One of the main departures in BCM from conventional modeling practice is that BCM does away with the "box and line" diagram style, using a window-icon representation instead. This is intended to present models in a more intuitively understandable way. BCM also introduces the idea of innate concept types: pre-defined categories into which mental concepts fall; this is intended to mimic human "common sense" understanding of the nature of people, organizations, places, activity, and so on. The introduction of innate concept types confers several benefits including an enhanced ability to select suitable pictures for concepts and easy reuse of existing concept definitions. It also permits "predictive modeling": the automated interpretation and completion of models, while they are being edited, based on the statistical likelihood of specific relationship configurations.

2.3. Evaluation of BCM Technique in Practice

The overall aim was to determine the relative usability of the psychologically-inspired BCM technique and the effectiveness of modelers using it, taking into account variables such as the modeler's prior experience. We chose to do this using a field experiment, paying careful attention to process, validity and reliability. In the absence of any established yardstick for the usability of conceptual modeling techniques we used object modeling as a control technique; BCM was therefore applied by modelers (expert and non-expert) alongside object modeling. Qualitative and quantitative data were gathered about each model and each modeling session using participant observation, questionnaires and interviews. Raw quantitative values were also derived by inspecting every version of each model and comparing every version with a final corrected version of the model (Table 4).

Table 4 Values from Measurement of Models

Value	Denoted by
Number of components	c
Number of finished components	c_f
Number of data items	c_a
Number of relationships	r
Number of correct relationships	r_c
Number of changes	m
Number of errors	m_e
Number of corrections	m_c
Modeling time	t

The experiment consisted of nineteen modeling exercises (Table 5) conducted by ten modelers (Table 6), each modeler working either alone or with a group of end users.

Table 5 Models

Business association administration	Mobile phone billing
College administration	Mobile phone network administration
Consulting administration	Mobile phone roaming
Data-related legislation	Purchase orders
Fund management	Retail distribution
Homeopathic medicine	Security standards
Human resources	Security/fraud management
IT help desk	Stock control
Legal group administration	Theatrical productions

Each model was developed using either BCM or object modeling. Most modelers produced only one model but eight were developed by a single (expert) modeler. Model development took approximately eighteen months and required negotiation and planning with potential subject organizations and modelers over the prior two-year period. Coding and analyzing the resulting model versions occupied approximately fourteen months. A secondary experiment was carried out for triangulation purposes (not described in this paper). More complete details of experimental procedure are set out in Appendix C (McGinnes 2000).

Table 6 Modelers

A	Company director/ex-lecturer
B	Ex-personnel assistant in a retail bank
C	Consultancy company administrator
D	Trainee systems analyst
E	Higher education college administrator (ex)
F	Audio-visual technician
G	Homeopathic medical practice administrator
H	Project manager
I	4th-year computer science student
J	Senior IT consultant

2.4. Measuring Effectiveness and Usability

Various quality measures for conceptual models have been proposed. Some focus on internal correctness (e.g. Hussain, Shamail and Awais 2004) while others are more concerned with matching mental concepts accurately (e.g. Siau and Tan 2005). After reviewing alternatives we decided to focus on two factors: *modeler effectiveness* and *modeling technique usability.* In order to evaluate model development, normalized measures were calculated from the raw values obtained from analysis of models; the normalized values include completeness, correctness, complexity, error rate and productivity. In Table 7, primed variables refer to values measured for the corrected version of each model.

Table 7 Values Calculated to Evaluate Model Development

Facet	Description	Formula
Completeness (q1)	The completeness of a model version is the percentage of model components and relationships in the finished version that are also present in this version.	$100 \times \left(\frac{c_f + r_c}{c' + r'} \right)$
Correctness (q2)	The correctness of a model version is the percentage of components in the current version that are present and defined correctly (i.e. defined in the same way as in the finished version).	$100 \times \left(\frac{c_f + r_c}{c + r} \right)$
Attribute ratio (ra)	The attribute ratio of a model version is the percentage of model components in the current version that are data items.	$\frac{c_a}{c}$
Volatility (v)	The volatility of a model version is the total number of changes in the current version relative to the total number of components and relationships.	$\frac{m}{(c+r)}$
Accuracy (a)	The accuracy of a model version is the percentage of changes in the current version that are corrections (i.e. changes that cause the model to become closer to its completed form).	$100 \times \left(\frac{m_c}{m} \right)$
Complexity (x)	The complexity of a model version is the average number of relationships for each model component.	$100 \times \left(\frac{r}{c} \right)$
Error rate (e)	The error rate for a modeler when producing a model is the proportion of changes made which were errors. A 'perfect' error rate is 0 (no errors at all) whilst a rate of 1 means that every change the modeler made was an error.	$\frac{\sum_v m_e}{c'}$
Productivity (p)	The productivity of a modeler when producing a model is the average number of finished (i.e. correct) business concepts produced per hour of modeling time.	$\frac{c'}{t}$

The measure *effectiveness* was calculated for each model using the formula below, where q_1 and q_2 are the completeness and correctness of the final uncorrected version of the model, p is the modeler's productivity, and e is the modeler's error rate.

$$Effectiveness = \frac{q_1 q_2 p(1-e)}{200}$$

The effectiveness value represents the modeler's ability to construct a complete model, without making mistakes, in good time. It is divided by 200 simply to produce a figure comparable to the other measures in this study. Complexity is not taken into account since it is not causally linked in any obvious way with model quality. Table 8 classifies effectiveness scores into ranges labeled for convenience as *excellent, good, poor* and *very poor*.

Table 8 Effectiveness Levels

Level	Score	Typical description
Excellent	>150	Correct and complete. Suitable for use in system design without refinement. Produced quickly and with few errors.
Good	101-150	Largely correct and complete, but typically usable only with further work (or produced relatively slowly).
Poor	51-100	Coherent but substantially incomplete and incorrect. Typically usable only as a 'first-cut' model.
Very poor	0-50	Incoherent and/or grossly incorrect. Produced very slowly and/or with many errors. Typically unusable even as a 'first-cut' model.

To calculate the usability of a modeling technique, we note that a modeler's effectiveness is governed by his or her own ability and skill together with the inherent usability of the method being employed. The contribution of modelers' abilities can be approximated by comparing effectiveness scores for different modelers who use the same modeling technique. This allows the impact of ability to be factored out to estimate the modeling technique's usability as follows:

Usability = Effectiveness – Ability

3. Results

Over one hundred separate model versions were recorded. It was found that novice modelers tended to produce somewhat smaller models using both techniques. However, model size was roughly consistent between the modeling techniques for each type of modeler (Table 9).

Table 9 Average Model size by Modeler's Experience Level and Method

Experience level	Average model size (no. of concepts and data items in model)	
	Object Modeling	BCM
Expert	70.2	81.7
Non-expert	46.2	44.8

3.1. Observations by Expert (Interview Notes)

The expert modeler found BCM mentally taxing (*"in respect of knowing where you are"*) but felt the participants (i.e. the end users) found it easier to understand. He attributed this to *"the simplified view—the graphic view is more accessible than a diagrammatic view that the participants are probably not familiar with"*. The participants were *"intuitively … more in touch with the picture—they are not faced with a big wiring diagram, they are digesting it in chunks"*.

With BCM, he found he had to call breaks to review the model. The participants went for coffee while the expert modeler checked the model in detail. *"If you have an object model on the board you can quickly see where the weak areas are, and where the relationships are. But with BCM you tend to follow a line from component to component, and not really know if you are missing another line somewhere else"*. While *"there is nothing wrong with calling regular breaks"*, he predicted that novice modeler would have difficulty finding areas in a model that needed work. The expert modeler reported that more was achieved with BCM in the modeling sessions than with object modeling. But he also felt that the BCM models probably needed more work after each session.

Despite initial reservations, the expert modeler ultimately expressed a preference for BCM over traditional object/data modeling. He reported that the quality of BCM models was better, which he attributed to improved understanding by the users. The act of classifying each concept (as a person, organization, document, etc.) aided understanding. It got the participants engaged in discussion and helped them clarify their ideas about what each concept actually meant. In this regard *"it almost doesn't matter what categories you have to choose from—it just helps to have categories so you have to have a discussion"*. Consequently, descriptions in BCM models were more accurate and the attributes better thought out: *"with BCM you are asking a more specific question—"what type is …?"—so you can get a more specific answer, not just general agreement to a description"*.

The expert modeler perceived other aspects of BCM as helpful. Being reminded of existing components was not particularly valuable for the modeler, but was useful for the participants. Choice of color and backgrounds assisted in recall and the ability to incorporate audio-visual material helped to increase arousal, providing *"a visual jab in the ribs"*. The expert modeler found the English-language interpretation an essential tool since it reduced the mental effort required to verbalize the model and made the model's meaning more concrete.

3.2. Questionnaire Responses

Analysis of questionnaires suggests that participants generally understood the modeling process using both techniques. For object modeling two specific issues were raised: *"(It is) difficult not to go off on a tangent and discuss outside (the) necessary area"* and *"(the exercise involved) attempting to develop something definite out of something that wasn't very definite"*. Respondents who had created a database before were more likely to judge their models as complete and correct and to correctly state the purpose of the modeling sessions, suggesting that creating a database provides insight into the reasons for modeling. For more details, please see Appendix D.

3.3. Completeness and Correctness

In the graphs that follow, each model has been allocated to a group according to the modeler's experience level and modeling technique (Table 10).

Table 10 Modeler Groups

Technique	Experience level		
	Novice (no experience)	Intermediate (some experience)	Expert (high experience)
Object modeling	Group 1	Group 2	Group 3
BCM	Group 4	Group 5	Group 6

In summary, the expert modeler produced 100% complete models using both BCM and object modeling. All BCM models were at least 80% complete; in contrast most object models produced by non-experts were less than 40% complete (Figure 3). Correctness showed a similar pattern, at over 80% for all but one BCM model but less than 25% for most object models produced by non-experts (average 22%). Object models exhibited significantly greater overall variability in completeness and correctness than BCM models.

For comparison, in the secondary study a different group of object modelers achieved average 43% completeness. These modelers were closest in experience level to group 2 in the main study, who also achieved average completeness of 43%. While the precise agreement in scores is obviously coincidental, we conclude that the results from the secondary study seem to support the completeness figures from the main study.

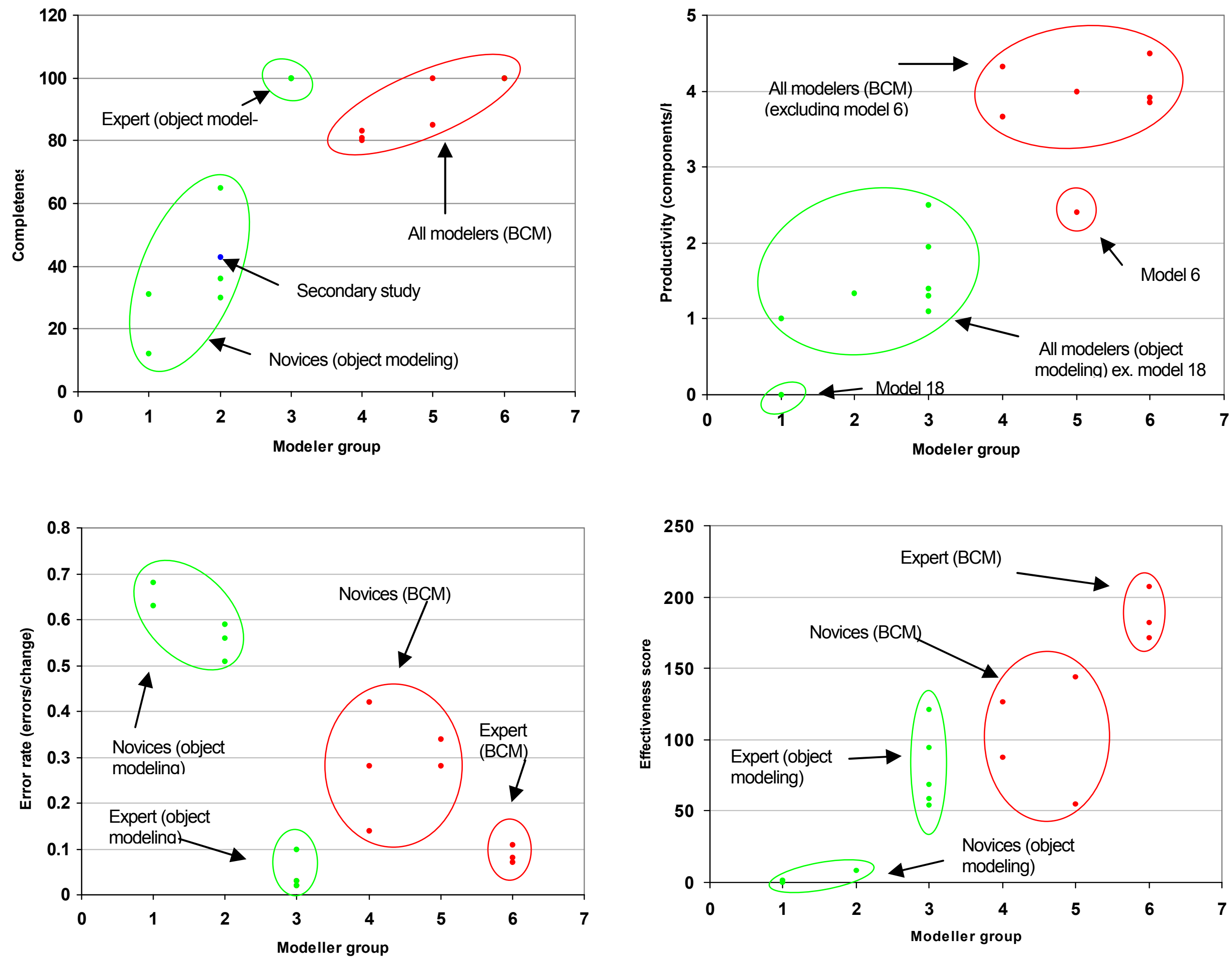
Productivity

Using BCM, the expert's average productivity was 147% greater than when using object modeling (Figure 3). For non-experts, average productivity was 366% greater than when using object modeling. Most BCM models were produced at a rate of over 3.5 concepts/hour; for object models the rate was less than 1.5 concepts/hour. With object modeling, the non-experts' productivity fell significantly short of that of the expert. But for BCM, productivity was more uniformly high for all modelers. Productivity figures were not available for three of the models since their total modeling time had not been recorded; therefore no effectiveness figures were calculated for these models (see below).

3.4. Errors

The expert modeler made few errors using either technique. Non-experts using object modeling made 0.59 errors per change; in contrast, non-experts using BCM made 0.29 errors per change (Figure 3). For most BCM models the figure was below 0.3; no object model achieved a value below 0.5 except those produced by the expert. In other words, nearly two-

Figure 3 Metrics by Modeler Group



thirds of the actions by non-expert object modelers were mistakes while less than one-third of actions by novice BCM modelers were mistakes.

Table 11 Error Frequencies

Error	Object modeling	BCM
Incorrect relationship added	73%	69%
Incorrect component added	20%	25%
Relationship wrongly removed	3%	2%
Cardinality altered wrongly	2%	2%
Component wrongly removed	1%	2%

The proportion of each type of error was very consistent between the two techniques (correlation .997). The most common type of error was creation of incorrect relationships—chiefly redundant relationships and relationships where participating concepts were wrongly chosen. Creation of incorrect (e.g. badly named or redundant) concepts occurred less frequently but was also significant. Other types of error occurred infrequently (Table 11).

3.5. Calculating Effectiveness and Usability

Based on the rubric in Table 8, the average effectiveness of modelers using object modeling was *very poor* (non-experts) or *poor* (expert) whilst that of modelers using BCM ranged from *good* (non-experts) to *excellent* (expert) (Figure 3). Looking at each model, the expert modeler was less effective using object modeling (four *poor* scores and one *good*) than with BCM (all *excellent*) (Table 12). This is not an indication that his object models were faulty but reflects the fact that they took longer to produce.

The "quality" measures used to calculate effectiveness—completeness, correctness and error rate—were strongly correlated (.95). Hence these measures may be regarded as facets of a single factor (Bryman 1990). For novices, productivity was correlated strongly with quality (.92) whereas, for the expert, quality was high irrespective of other factors. Complexity was inversely correlated with all other measures, especially for the novice modelers (.73).

Table 12 Summary of Effectiveness Levels Achieved

Method	Modeler's experience	Effectiveness (no. of models)			
		Very poor	Poor	Good	Excellent
BCM	None		1	1	
	Some		1	1	
	High				3
Object modeling	None	2			
	Some	2			
	High		4	1	

Table 13 calculates the differences between effectiveness values. From this analysis, the contribution of experience amounts to approximately 34-38% of an average modeler's overall effectiveness. The contribution of the model technique's usability is 46-50%, or about 33% greater than the contribution of experience (please note the relatively large margins of error for these figures, however).

Table 13 Average Effectiveness (as % of nominal maximum effectiveness)

Method	Novice modelers	Expert modeler	Difference (experience)
Object modeling	2% (±02)	40% (±14)	38% (±14)
BCM	52% (±20)	86% (±25)	34% (±25)
Difference (usability of technique)	50% (±20)	46% (±25)	

3.6. Discussion

The results suggest that the introduction of psychological ideas in BCM had a significant impact on effectiveness, which was comparable to or exceeded the contribution of experience. BCM allowed non-expert modelers to emulate the performance of an expert modeler who was using industry-standard techniques. It allowed all modelers, regardless of experience, to be roughly two to five times more productive.

The results are striking, but are they meaningful? There are probably alternative ways of calculating effectiveness that would give different results. In particular, our measures (of completeness, correctness, error rate and productivity) seem too coarse to effectively register variations in the expert's performance other than productivity. This may help explain why the expert modeler was found to be somewhat less effective when using object modeling than non-experts using BCM. However, it seems likely that any method based on the same four factors would be subject to the same trends and would give broadly similar results. Overall, the quantitative results tally with qualitative observations of modelers during the experiment. This is especially true for non-expert modelers, where a stark difference in performance was observed between those using BCM and those using object modeling.

4. Analysis

A motive behind this research is to find out if the level of expertise needed to perform conceptual modeling can be reduced. It seems that the answer is "yes"; under the right circumstances, novice modelers with little or no prior experience can perform as well as an expert modeler with extensive experience. Despite the limited sample size, quite compelling results were obtained.

4.1. Helping Novices to Emulate Experts?

Psychology tells us that experts have mental frameworks for problem-solving and internalized skills that can be applied without the need for conscious thought. Non-experts lack both. In recent years it has been claimed that novices can emulate expert behaviors successfully (Tosey 2005). We approached the same kind of goal using three main strategies: (a) by adjusting the content of models to obtain a better match with mental concepts (e.g. by adding innate concepts), (b) by representing models in a way that people might understand more easily, and (c) by trying to make the modeling process simpler and easier (Backhouse 1988; McGinnes 1994). The aim was to empower non-experts by giving them predefined frameworks and by allowing them to use skills they already have, such as visual recognition.

Surveying the literature on conceptual modeling, one might be forgiven for thinking that all modelers are specialists who embark on modeling only once they are armed with the requisite skills. The focus has historically been on issues like formality, economy of representation and rigor rather than simplicity or ease of use (Gregory 1995; Herbst 2000). But we suspect that a good proportion of conceptual models are produced by non-experts, some explicitly and others implicitly during the design of end-user systems using products such as Microsoft Access and Lotus Notes. With constant change and growth in the IT industry, demand for trained staff often outstrips supply, yet information systems must still (and will) be developed.

Another implicit assumption is that modelers will continue to work on models until they are correct, before implementing them in databases and applications. In this study the non-expert modelers were typically unaware of their mistakes and therefore did not fix them. In practice, IT professionals are subject to deadlines and resource constraints which can prevent them from spending sufficient time on design. This is one reason why production systems often contain significant structural flaws. In this study, the novice modelers using BCM produced models that were 80% complete and correct. We speculate that this level of quality may not be far from current professional practice.

4.2. Emergent Patterns of Expertise

During detailed analysis of the models, three distinct patterns of model evolution were observed (**Error! Reference source not found.**).

Pattern A: Expert modeler (both techniques): This pattern appeared for all models produced by the expert modeler. It demonstrates the kind of behavior that one might expect of someone who is experienced and capable. The expert goes straight to a correct solution with confidence and efficiency; accuracy, completeness and correctness quickly climb to 100%. The error rate tends rapidly towards zero. Complexity is high but this seems not to affect performance and declines slightly as the model is completed satisfactorily.

Pattern B: Non-expert modeler (object modeling only): This pattern appeared for object models produced by non-experts; it presents a picture of a modeler who is unsure of what to do. The modeler makes a series of mistakes and seems unable or unwilling to correct them. Errors accumulate until the modeler either gives up or wrongly judges the model to be complete. The model quickly becomes over-complex and accuracy, completeness and correctness remain low. The model fails to reach a satisfactory state.

Pattern C: Non-expert modeler (BCM only): This pattern appeared for all models produced by non-experts using BCM. It is similar to pattern A (expert modeler) in that the model reaches a satisfactory state. However, pattern C appears to show evidence of learning and improvement while the modeling progresses. After an initial period of growth the number of errors remains roughly stable in a "plateau" phase. Most of the errors are incorrect relationships. Eventually the modeler fixes the errors and goes on to complete the model. Model complexity initially rises quickly but the model does not become over-complex. Completeness and correctness show overall increasing trends. The modeler's accuracy fluctuates but improves as the modeling progresses.

Comparison of patterns: The typical profile for each type of modeler is summarized in Table 14. We note that patterns B and C, for non-expert modelers using object modeling and BCM respectively, are very different. In a typical example of pattern B (object modeling) the novice modeler created a model that was not structured meaningfully. Reasonably correct classes were identified but few, if any, relationships were correct. Models of this nature were generally not completed, as stated above. In contrast, pattern C models (BCM) did proceed to satisfactory completion.

Table 14 Summary of Typical Performance for Each Type of Modeler

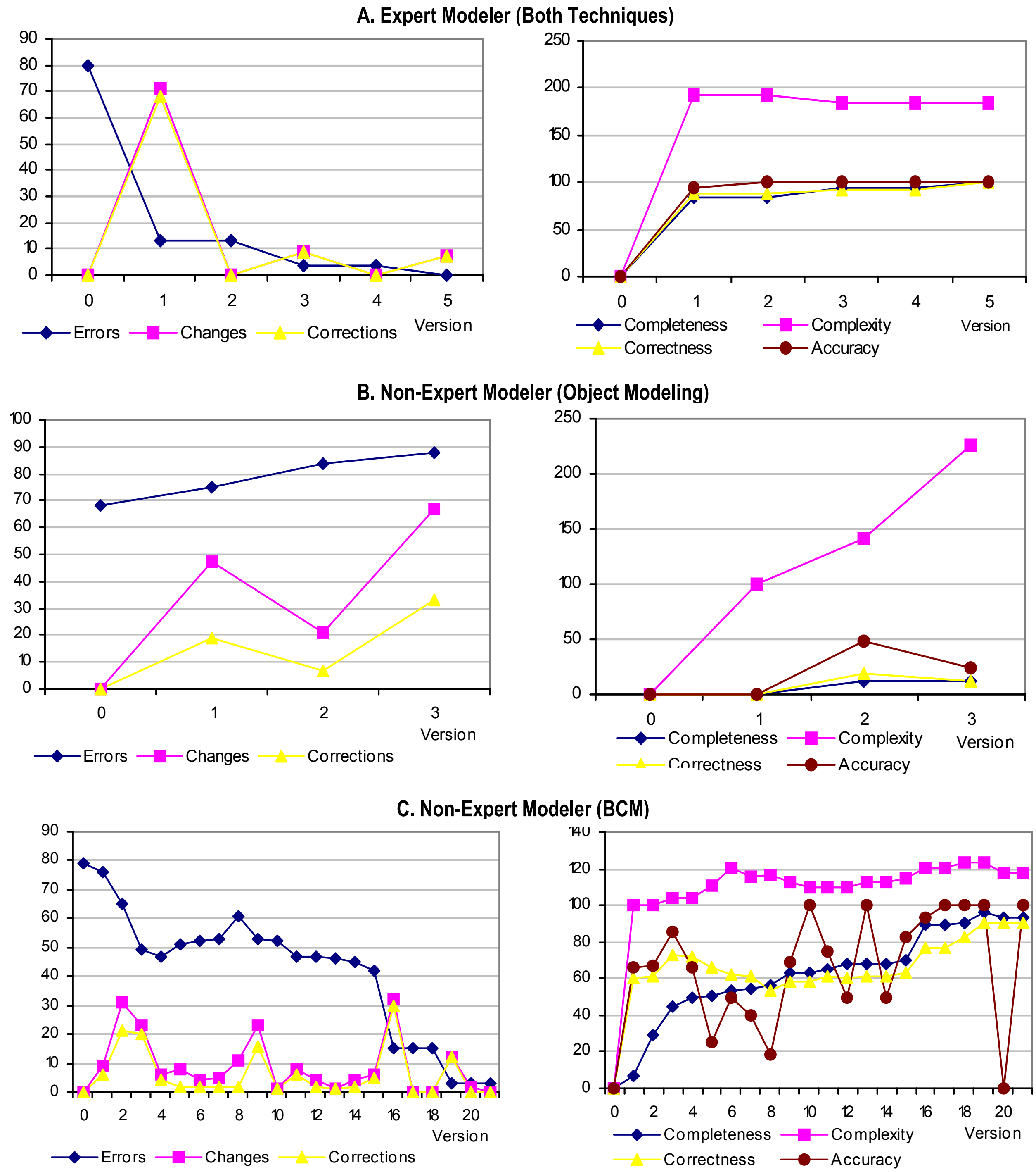
	Object modeling		BCM	
	Expert	Non-expert	Expert	Non-expert
Pattern:	A	B	A	C
Productivity:	Low	Very low	Very high	High
Effectiveness:	Poor	Very poor	Excellent	Good

The similarities between patterns A (expert modeler) and C (non-expert using BCM) suggest that BCM helped non-expert modelers to behave like the expert in some important ways. An expert can go directly to a good solution; as we might expect, the curve for the expert (pattern A) shows a continuously decreasing error level. But a novice cannot behave identically to an expert since he or she typically does not know in advance how best to model. The non-expert modelers using object modeling accumulated errors and were unable to correct them (pattern B). Novice modelers using BCM also made errors (pattern C), especially in the early stages, but were able to recognize their errors and eventually correct them. So it seems that what is crucial is not whether a modeler makes mistakes, but whether he or she is ultimately able to identify and correct those mistakes.

4.3. Other Evidence

Other observations support the hypothesis that BCM helped non-experts to emulate expert behavior. All non-expert modelers defined fewer relationships per concept than the expert. But the ratio for non-expert modelers using BCM was closer to that of the expert.

Figure 4 Evolutionary Patterns



Similarly, non-experts using object modeling tended to make proportionally many more changes than the expert modeler, whereas for BCM the number of changes they made was (in most cases) closer to that of the expert. Complexity seems to be particularly problematic for non-expert modelers; those using object modeling were unable to prevent the complexity of their models from increasing in an unbounded way. In contrast, non-expert modelers using BCM were able to keep the complexity of their models to manageable levels.

5. Conclusions

The results of this study seem to support the use of psychologically-inspired innovations in conceptual modeling both for experts and for non-experts. Using BCM, people with little or no prior experience produced conceptual models of near-expert quality at least as quickly as an expert. The models were smaller and less complex than those produced by the expert but still of a realistic and useful size. For the expert modeler, BCM allowed significant productivity improvements.

We would caution against too literal an interpretation of the experimental results. Although efforts were made to control the study, in the scientific sense, it was subject to all of the "messiness" of real-life business situations; subjectivity was unavoidable, for instance, in the correction of models. It could also be that our measurements lacked the sensitivity to detect small but crucial differences in model structure which have a large impact on the downstream usefulness of a model. However, in general we can say that evidence supports the idea of applying psychological principles to conceptual modeling. The results do not tell us specifically which factors, whether psychological innovations or other factors, led to the observed benefits. However, we can speculate on the likely causes of performance improvement by comparing qualitative observations with the quantitative measurements. Some comments in this direction are given below.

5.1. Reflections

It appears that the non-experts using BCM were able to learn by trial and error. They frequently consulted the English-language interpretation and experimentation was easy using the BCM tool; hence the "plateau phase" visible in pattern C. They made mistakes (mainly wrong relationships) but identified and fixed them. In contrast, non-expert object modelers had no such support and redrafting was onerous. They failed to learn or correct their mistakes, as reflected in pattern B.

To structure a conceptual model correctly requires one to employ logic. Experiments have suggested that logic and causality are not innate modes of thought and may be a 'syllogistic game'. *"In highly-industrialized Western societies, people are trained to prove arguments about reality on the basis of representational propositions. In less industrial societies … the form of proof is tied more directly to sensory impression"* (Solso, MacLin and MacLin 2004). In other words, people do not necessarily think analytically. Indeed, it can be exhausting to think in an analytical mode continuously. All non-expert modelers in this study lost concentration and forgot the meaning of model constructs from time to time. The idea of a model as a series of logical propositions was clearly foreign to several of them. Perhaps expert modelers are more at home with the analytical mode of thought and can switch between formal and intuitive modes at will, using intuitive thinking for creation and discovery, and formal thinking when checking and correcting meaning. If so, it might be better if non-experts were not forced into a continuously analytical mode of thinking; BCM's use of recognizable images may have allowed the non-expert modelers to use intuitive interpretation for at least part of the time.

BCM inherently restricts the view one has of a model since each concept has its own window. Participants were *"intuitively … digesting it in chunks"*. The expert modeler's language is telling since he was apparently unaware of the psychological concept "chunking". The restricted view helped non-expert modelers to pay attention to a single concept at a time. Presumably this was beneficial since their limited cognitive resources could be selectively focused. The natural-language interpretation in BCM can be viewed for a single concept and this was undoubtedly useful to modelers; they were able to see a brief, uncluttered summary of each concept's meaning without other potentially confusing information (this is not as simple to achieve as it might sound, when we remember that each concept is defined *in terms of* other concepts). The restricted view in BCM also created a need for navigation between windows, which seemed beneficial for non-experts. It provided a physical and intuitive way for modelers to remember where they had placed concepts. Non-expert object modelers suffered from a lack of focus, often attempting to read their model as if its meaning could be stated in a single sentence—which is incorrect and was one reason why they misinterpreted their diagrams.

The expert modeler formulated generic concepts (supertypes) as a way of factoring out common properties when using both techniques. Non-experts did not do this; they defined concepts instead at an everyday level of generality, despite support in both BCM and object modeling for this construct. Since the use of supertypes can be beneficial in a conceptual model, we speculate that additional support in BCM could help non-expert modelers use supertypes more effectively.

The predefined set of innate types in BCM helped to remind non-expert modelers to look for concepts under each heading. Non-expert object modelers had no such checklist and often missed important aspects altogether (e.g. identifying people and organizations but failing to identify important activities). In BCM, non-experts seemed most at home with the more concrete types: *person, organization, document, place* and (to a lesser extent) *system.* Other innate types, particularly *physical object, category* and *conceptual object*, seemed too general to be understood well; they were used indiscriminately by the non-experts. The innate type *activity* also seemed problematic for some non-experts. In object modeling, non-experts seemed to have trouble grasping the (very generic) concepts *class* and *association*. All of this accords with psychological evidence that people are typically most comfortable with categories pitched at an everyday level of generality. It suggests that the more generic innate types could usefully be refined to make them less generic.

BCM is designed for tool support and cannot easily be practiced without it. In this study we decided to use the BCM tool *in* modeling sessions but object modeling tools *outside of* sessions. The primary reason for this was that we felt that non-experts had enough to do in learning how to model without also having to master a traditional CASE tool. It is arguable that this decision introduced experimental bias; differences in modeling performance may have been due to the absence of a modeling tool in some sessions. Intuitively, however, this seems unlikely; our experiences with non-experts suggest that they would have been even more "at sea" if confronted by a typical CASE tool without the opportunity for proper training. The literature tends to support this view (Jarzahek and Huang 1998; Kline, Seffah, Javahery, Donayee and Rilling 2002).

5.2. Future Research

We have circumstantial evidence that the psychological innovations led to improvement in modeling effectiveness but we have no direct evidence of each factor's impact. As a priority we would like to look more closely at each innovation to determine whether, and how much, it contributes to the observed benefits. More specific knowledge will allow BCM to be tuned and may suggest further innovations; it may also allow us to discard certain aspects of BCM if they provide no particular benefit in themselves. This is desirable from a pragmatic standpoint since we would wish to obtain the maximum improvement with the minimum of difference from established best practice (i.e. object and data modeling).

In this study, non-experts using BCM produced models that were 80% complete and correct. How can we improve on this figure? One route would start with a more detailed analysis of the specific errors made by non-experts. It may be that certain clichés (repeated sequences of actions) occur; identifying these could suggest how to encourage "correct" modeling and avoid the error scenarios. Other ideas include creating a cookbook method to support the modeling technique, enhancing tool support with wizards for common business scenarios, and providing a stronger conceptual framework through more prescriptive model structures.

As a starting point, we know that non-expert modelers have particular difficulty in modeling relationships (Wand, Storey and Weber 1999). Over two-thirds of errors made in this study by modelers involved incorrect relationships, many of which were redundant. Therefore some support for correct modeling of relationships would be likely to confer benefit. Potentially redundant relationships exist where two concepts are explicitly or implicitly linked in more than one way. Perhaps BCM could help to avoid this type of error by identifying and highlighting potentially redundant relationships, thereby assisting the modeler in resolving them.

Another way of helping non-experts with relationships relies on the fact that one concept reminds us of another. This reminding typically indicates the presence of a relationship of interest between the two concepts. It is useful therefore to capture the linkage automatically, sparing the modeler the effort of remembering to create a relationship manually. This can be done if the BCM tool automatically displays an empty window for each new concept after it has been defined. The next concept that is defined will automatically be placed in the previous concept's window, creating an association between the two concepts. This "depth-first" approach has been tried and yields apparently acceptable results; however, we have yet to measure whether it results in more correct structures than the default approach, where (typically) several concepts are identified before any are related.

Why did non-experts using BCM succeed in correcting their mistakes while those using object modeling failed to do so? One factor may have been confidence, perhaps brought about by the relative ease or difficulty of making changes and obtaining feedback. The concept of *self-efficacy* has previously been linked with success in learning conceptual modeling skills; it hinges on feedback and early success (Bandura 1997): *"if individuals were successful in the past they were more likely to believe they could accomplish similar tasks in the future"* (Ryan, Bordoloi and Harrison 2000). We speculate that BCM users were given confidence by the positive feedback they got from the BCM tool and were therefore more incline to persevere.

Statistical analysis of models shows that certain patterns occur repeatedly in the relationships between concepts. For example, each pair of innate types occurs most frequently with a specific set of cardinalities. To take advantage of this fact the BCM tool defaults to the most likely relationship if cardinalities are left unspecified. The "most likely" relationships have been calculated in advance by examining a large set of models. This is *predictive modeling*; the use of probabilities to interpret and complete certain aspects of a model while it is being edited, saving the modeler the effort of specifying them explicitly. Predictive modeling has been implemented in the BCM tool and is apparently useful. It would be enlightening to test how often the assumed relationships are correct and to determine what other aspects could also be defaulted.

Although this study did not encompass application development, BCM can be used to build software applications directly from models and has successfully been used in this way on a variety of commercial projects. Further research could assess the use of BCM in model-driven application development by non-experts such as end users. This may bring us closer to the goal of collapsing the development and use of applications into a single process. Although current thinking on software project structure envisages application development as an iterative process, the development and use of software applications are still considered as separate activities that must be done by different people. With end-user tools based on the BCM principles we may be able to achieve a closer integration between the two, making a distinction in task and role less necessary.

Conceptual reuse is also a compelling area for further research. This is the idea of building new models (or applications) from existing concept definitions. When using BCM it is rarely necessary to construct a model from scratch since the BCM tool incorporates features that make it relatively easy to reuse existing concept definitions—or even whole models—in new models. There are two main advantages in this approach: firstly one does not need to reinvent the wheel and therefore much time is saved; secondly, existing concept definitions have already passed through a review process and are therefore (one would hope) well thought-out. The downside is that reused concepts may be inappropriate in detail. However, the strongly visual aspect of BCM models, and the availability of the English-language interpretation and form preview functions, seem to make it easier to determine if existing concept definitions are suitable for new needs, and the modeler can easily alter an existing definition to suit a new situation.

The expert modeler pointed out that BCM lacks the ability to provide an overview of a model; there is no view that shows everything *in context*. He predicted that non-experts would have trouble finding the parts of a model that needed work. Intuitively he seems correct; however, we did not see any evidence that a lack of "big picture" hindered the modelers. Perhaps the benefits of information hiding outweighed the disadvantages of not seeing a big picture. Although the expert's prediction was not borne out in the evidence, it may nevertheless be that a "whole model" view could be beneficial under certain circumstances; one possibility that has been considered is an advanced model navigation facility which allows models to be visualized and traversed in three-dimensional space.

5.3. Conclusions

The synthesis of new techniques and tools has in recent years been somewhat out of favor in the IS research world. Yet, in software, nothing is given; we are free to imagine alternative ways of doing things and limited only by our ability to formulate them. There is great scope in IS research for creativity, and creative research can make a genuinely positive contribution to practice. It is gratifying to have the opportunity to present the results of design-oriented research to an IS readership.

This study has demonstrated ways of improving both the quality of conceptual models and the performance of conceptual modelers. We have seen that conceptual modeling is not a black art but a relatively predictable activity where the outcome depends largely on controllable factors. For non-expert modelers, the choice of modeling technique has a significant impact on model quality, making the difference between results that are adequate and results that are essentially unusable. For both expert and non-expert modelers, the choice of modeling technique can significantly improve productivity.

We conclude that there is great scope for improvement in the usability of conceptual modeling techniques. Conceptual modeling is practiced by significant numbers of non-experts, including less experienced IT professionals and end-users with products such as Microsoft Access. The complexity of our existing modeling techniques presents a barrier to these people. Today's conceptual modeling techniques neglect our innate, automatic visual recognition capabilities and they place unrealistic demands on memory. They seek to capture end users' knowledge and business concepts but fail to offer any clear correspondence with mental models. They are typically hard for non-experts to understand and practicing them requires great mental effort.

The good news is that conceptual modeling can be simpler, quicker and less error-prone for both experts and non-experts. Psychology, particularly in the areas of cognition and group dynamics, offers much to aid our thinking about conceptual modeling—provided that we can apply what we know about how the world is perceived and how meaning is created within the mind.

6. References

Adams, C., Avison, D. "Dangers Inherent in the Use of Techniques: Identifying Framing Influences", *Information Technology & People* (16:2), June 2003, pp. 203-234.

Alexander, I. F., Stevens, R. *Writing Better Requirements*, Addison Wesley, London, 2002.

Backhouse, J. "Simpler and Richer Cognitive Models for ISAD", *IFIP International Workshop on Human Factors of Information Systems Analysis and Design*, The London School of Economics and Political Science, London, 1988.

Bandura, A. *Self-Efficacy: The Exercise of Control*, W.H. Freeman & Company, New York, 1997.

Bannert, M. "Managing Cognitive Load: Recent Trends in Cognitive Load Theory", *Learning and Instruction* (12:1), 2002, pp. 139-146.

Bansler, J. P., Bødker, K. "A Reappraisal of Structured Analysis: Design in an Organizational Context", *ACM Transactions on Information Systems* (11:2), April 1993, pp. 165-193.

Bell, A. E. "UML Fever", *ACM Queue* (3:2), March 2005, pp. 48-56.

Berger, P. L., Luckman, T. The Social Construction of Reality: A Treatise in the Sociology of Knowledge, Anchor Books, New York, 1966.

Bolloju, N. "Improving the Quality of Business Object Models Using Collaboration Patterns," *Communications of the ACM* **(**47:7), July 2004, pp. 81-86.

Bryman, A. Research Methods and Organization Studies, Unwin Hyman, London, 1990.

Burek, P. "Essentialized Conceptual Structures in Ontology Modeling", in: Rajiv Khosla, Robert J. Howlett, Lakhmi C. Jain (Eds.) *Knowledge-Based Intelligent Information and Engineering Systems: 9th International Conference, KES 2005, Melbourne, Australia, September 14-16, 2005, Proceedings, Part II*, Lecture Notes in Computer Science, Volume 3682, August 2005, Springer, Berlin/Heidelberg, pp. 880-886.

Burrell, G., Morgan, G. Sociological Paradigms and Organizational Analysis: Elements of the Sociology of Corporate Life, Heinemann, London, 1979.

Chen, P. "From Ancient Egyptian Language to Future Conceptual Modeling" in: P. Chen, J. Akoka, H. Kangassalo, B. Thalheim (Eds.) *Conceptual Modeling: Current Issues and Future Directions*, Lecture Notes in Computer Science, Volume 1565, January 1999, Springer, Berlin/Heidelberg, pp. 57-66.

Dennis, A. R., Hayes, G. S., Daniels, Jr. R. M. "Business Process Modeling with Group Support Systems", *Journal of Management Information Systems* (15:4), Spring 1999, pp. 115-143.

Dennis, A. R., Pinsonneault, A., Hilmer, K. M., Barki, H., Gallupe, R. B., Huber, M., Bellavance, F. "Patterns in Electronic Brainstorming: the Effects of Synergy and Social Loafing on Group Idea Generation", *International Journal of e-Collaboration*, (1:4), October 2005, pp. 38-57.

Esculier, C., Friesen, O. "Conceptual Modeling: A Critical Survey and a Few Perspectives" in: *Proceedings of the 1995 IEEE Fourteenth Annual International Phoenix Conference on Computers and Communications*, IEEE Computer Society Press, Los Alamitos, California, March 1995, pp. 319-325.

Evermann, J. "Towards a Cognitive Foundation for Knowledge Representation", *Information Systems Journal* (15:2), April 2005, pp. 147-178.

Eysenck, M. W., Keane, M. T. *Cognitive Psychology: A Student's Handbook*, 5th Edition, Psychology Press, Hove, UK, June 2005.

Gobet, F. "Chunking Models of Expertise: Implications for Education", *Applied Cognitive Psychology* (19:2), March 2005, pp. 183-204.

Goldstein, E. B. Cognitive Psychology: Connecting Mind, Research and Everyday Experience, Wadsworth Publishing, Belmont, CA, 2005.

Gregory, F. H. *Mapping Information Systems Onto the Real World*, City University of Hong Kong, Department of Information Systems, Working Paper No. WP 95/01, 1995.

Hammer, M., Champy, J. Reengineering the Corporation: a Manifesto for Business Revolution, HarperBusiness, New York, December 2003.

Herbst, H. *Business Rule-Oriented Conceptual Modeling*, Physica-Verlag, Heidelberg, November 2000.

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

Hitchman, S. "Using DEKAF to Understand Data Modeling in the Practitioner Domain", *European Journal of Information Systems* (6:3), September 1997, pp. 181-189.

Hitchman, S. "The Entity Relationship Model and Practical Data Modelling", *Journal of Conceptual Modeling* (31), March 2004.

Jarzahek, S., Huang, R. "The Case for User-Centered CASE Tools", *Communications of the ACM*, (41:8), August 1998, pp. 93-99.

Johnson-Laird, P. N. "Mental Models and Thought" in: Holyoak, K. J. and Morrison, R.J. (Eds.) *The Cambridge Handbook of Thinking and Reasoning*, Cambridge University Press, Cambridge, 2005, pp. 185-208.

Jurisica, I., Mylopoulos, J., Yu, E. "Using Ontologies for Knowledge Management: An Information Systems Perspective", *Knowledge and Information Systems*, (6:4), July 2004, pp. 380-401.

Justice, T., Jamieson, D. *The Facilitator's Fieldbook*, 2nd Edition, AMACOM, American Management Association, New York, June 2006.

Kline, R., Seffah, A., Javahery, H., Donayee, M., Rilling, J. "Quantifying Developer Experiences via Heuristic and Psychometric Evaluation" in: *Proceedings of the IEEE 2002 Symposia on Human Centric Computing Languages and Environments (HCC'02)*, IEE Computer Society, 2002.

Leung, F., Bolloju, N. "Analyzing the Quality of Domain Models Developed by Novice Systems Analysts" in: *Proceedings of the 38th Annual Hawaii International Conference on System Sciences (HICSS'05)*, 2005, pp. 188b.

LeVasseur, J. J. "The Problem of Bracketing in Phenomenology", *Qualitative Health Research* (13:3), March 2003, pp. 408-420.

Loucopoulos, P., Zicari, R. *Conceptual Modeling, Databases and CASE*, John Wiley, Chichester, 1992.

McGinnes, S. "CASE Support for Collaborative Modeling: Re-engineering Conceptual Modeling Techniques to Exploit the Potential of CASE Tools", *Software Engineering Journal* (9:4), July 1994, pp. 183-189.

McGinnes, S. *Conceptual Modeling: A Psychological Approach*, Doctoral Thesis, Department of Information Systems, London School of Economics, University of London, 2000.

McGinnes, S., Amos, J. "Accelerated Business Concept Modeling: Combining User Interface Design with Object Modeling" in: *Object Modeling and User Interface Design: Designing Interactive Systems*, Addison-Wesley, Boston, MA, 2001, pp. 3-36.

Medin, D. L., Ross, B. H., Markman, A. B. *Cognitive Psychology*, John Wiley & Sons, Chichester, 2004.

Moody, D. L. "Cognitive Load Effects on End User Understanding of Conceptual Models: An Experimental Analysis", *Lecture Notes in Computer Science*, Volume 3255, Springer, Berlin/Heidelberg, 2004, pp. 129-143.

Nielsen, J., Loranger, H. *Prioritizing Web Usability*, New Riders Press, Thousand Oaks, CA, 2006.

Olle, T., Hagelstein, J., Macdonald, I., Rolland, C., Sol, H., van Assche, F., Verrijn-Stuart, A. *Information Systems Methodologies: A Framework for Understanding* (2nd ed.), Addison Wesley, Reading, Massachusetts, 1991.

Orlikowski, W., Iacono, C. "Desperately Seeking the "IT" in IT Research - A Call to Theorizing the IT Artifact" *Information Systems Research* (12:2), June 2001, pp. 121-134.

Ramesh, V., Dennis, A. R. "The Object-Oriented Team: Lessons for Virtual Teams from Global Software Development" in: *Proceedings of the 35th Annual Hawaii International Conference on System Sciences, 7-10 Jan. 2002*, 2002, pp. 212- 221.

Ramesh, V., Parsons, J., Browne, G. J. "What Is The Role of Cognition in Conceptual Modeling? A Report on the First Workshop on Cognition and Conceptual Modeling", *Lecture Notes in Computer Science*, Volume 1565, Springer, Berlin/Heidelberg, 1999, pp. 272-280.

Remenyi, D., Williams, B. "The Nature of Research: Qualitative or Quantitative, Narrative, or Paradigmatic?", *Information Systems Journal* (6:2), April 1996, pp. 131-146.

Rosenberg, D., Scott, K. Applying Use Case Driven Object Modeling with UML: An Annotated e-Commerce Example, Addison Wesley Professional, Harlow, 2001.

Ryan, S. D., Bordoloi, B., Harrison, D. A. "Acquiring Conceptual Data Modeling Skills: The Effect of Cooperative Learning and Self-Efficacy on Learning Outcomes", *ACM SIGMIS Database* (31:4), September 2000, pp. 9-24.

Schank, R. C. *Dynamic Memory Revisited*, Cambridge University Press, Cambridge, 1999.

Schweizer, K. "Preattentive Processing and Cognitive Ability", *Intelligence* (29:2), Match-April 2001, pp. 169-186.

Shanks, G., Tansley, E., Weber, R. "Using Ontology to Validate Conceptual Models", *Communications of the ACM* (46:10), October 2003, pp. 85-89.

Shanks, G., Tansley, E., Weber, R. "Representing Composites in Conceptual Modeling", *Communications of the ACM* (47:7), July 2004, p. 77-80.

Siau, K., Tan, X. "Improving the Quality of Conceptual Modeling Using Cognitive Mapping Techniques", *Data & Knowledge Engineering* (55:3), December 2005, pp. 343 – 365.

Siau, K. "Human-Computer Interaction: The Effect of Application Domain Knowledge on Icon Visualization", *The Journal of Computer Information Systems* (45:3), Spring 2005, pp. 53-62.

Simon, H.A. *The Sciences of the Artificial*, 3rd Edition, MIT Press, Cambridge, MA, 1996.

Solso, R. L., MacLin, M. K., MacLin, O. H. *Cognitive Psychology* (7th ed.), Allyn and Bacon, Boston, 2004.

Storey, V. C., Chiang, R. H. L., Dey, D., Goldstein, R. C., Sudaresan, S. "Database Design With Common Sense Business Reasoning and Learning", *ACM Transactions on Database Systems* (22:4), December 1997, pp. 471-512.

Sowa, J. F. Knowledge Representation: Logical, Philosophical, and Computational Foundations, Brooks Cole Publishing Co., Pacific Grove, CA, 2000.

Topi, H., Ramesh, V. "Human Factors Research on Data Modeling: A Review of Prior Research, An Extended Framework and Future Research Directions", *Journal of Database Management* 13(2), April-June 2002, pp. 3-19.

Tosey, P. "Mapping Transformative Learning: The Potential of Neuro-Linguistic Programming", *Journal of Transformative Education* 3(2), April 2005, pp. 140-167.

Vaishnavi, V., Kuechler, W. "Design Research in Information Systems" last updated January 18, 2006. URL: http://www.isworld.org/Researchdesign/drisISworld.htm, Authors e-mail: vvaishna@gsu.edu kuechler@unr.edu, 2006.

Venable, J. R. "Teaching Novice Conceptual Data modelers to Become Experts" in: *Proceedings of the International Conference on Software Engineering: Education and Practice (SE:EP '96)*, IEEE Computer Society Press, pp. 50-56, 1996.

Veres, C., Mansson, G. "Cognition and Modeling: Foundations for Research and Practice", *Journal of Information Technology Theory and Application* 7(1), 2005, pp. 93-104.

Wand, Y., Storey, V. C., Weber, R. "An Ontological Analysis of the Relationship Construct in Conceptual Modeling", *ACM Transactions on Database Systems* 24(4), December 1999, pp. 494-528.

Wilson, B. Soft Systems Methodology: Conceptual Model Building and Its Contribution, John Wiley & Sons, Chichester, 2001.

Zander, A. *Making Groups Effective*, Jossey-Bass, San Francisco, 1994.

Zmud, R. “Editor’s Comments”, *MIS Quarterly* (21:2), June 1997, pp. xxi-xxii.

7. Appendix A: Selected Psychological Principles

Twenty-nine psychological principles were formulated in total. The following set of ten principles is illustrative.

7.1. Principle 1: Preattentive Processing

Maximize Bandwidth Using Visuals. Preattentive processing is the unconscious, automatic processing of sensory inputs (Schweizer 2001). Graphs and rapid visual searching rely on our capability for preattentive processing to convey complex data efficiently, using variation in color, shape and position. Preattentive processing is effectively instantaneous and requires little cognitive effort. But it has limitations; important visual features cannot be varied *in combination* since this forces the use of attentive (i.e. conscious, analytical) processing. Figure 6 illustrates how the brain gathers information more easily when it is presented in visual form. Perhaps the potential for conceptual modeling lies in allowing these capabilities to be used in the interpretation of models. To permit preattentive processing, conceptual models would need to be expressed using visual constructions that non-experts can interpret automatically and unconsciously.

Figure 5 Translating Between Conceptual and Mental Models

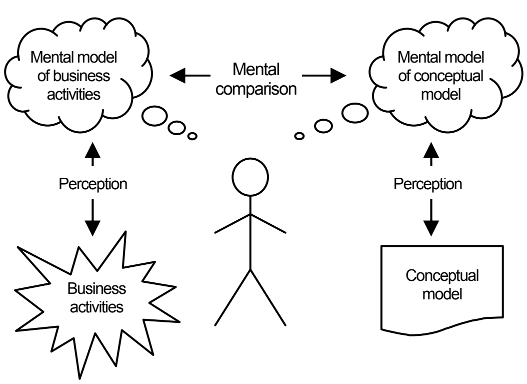
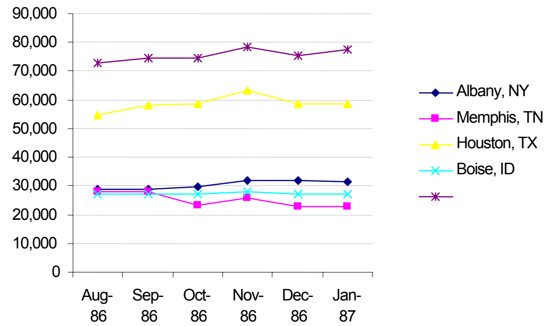


Figure 6 Information Presented in both Numerical and Graphical Form

	Aug-86	Sep-86	Oct-86	Nov-86	Dec-86	Jan-87
Albany, NY	$28,675	$28,675	$29,575	$31,875	$31,675	$31,650
Memphis, TN	$28,200	$28,200	$23,400	$25,900	$22,900	$22,900
Houston, TX	$54,500	$58,000	$58,500	$63,500	$58,500	$58,500
Boise, ID	$27,250	$27,250	$27,250	$27,900	$27,250	$27,250
Minneapolis, MN	$72,950	$74,500	$74,500	$78,500	$75,425	$77,525



7.2. Principle 2: Isomorphism:

Reduce Cognitive Load by Matching Conceptual and Mental Models. Conceptual models help the software designer to gain an accurate mental model of the business, so that the resulting systems fit the organization well; inaccurate models can be expensive and damaging. To verify a conceptual model we must form a mental model of it using familiar concepts (Solso, MacLin and MacLin 2004). We are then free to compare mental and conceptual models (Figure 5). Discrepancies lead us to revise either the conceptual model, our interpretation of it, or our mental model of the business.

An expert performs these mental gymnastics both rapidly and unconsciously; the skill has become internalized. But novices must attend to each part of a model separately, just as a learner car driver remembers how to change gears as a sequence of discrete actions. The great effort needed for this kind of thinking has been likened to *“using a helicopter to hold up a clothes line”* (Sowa 2000). A model may be expressed in terms so unfamiliar that too much cognitive effort is needed to understand it, as anyone who has struggled to understand a badly-written specification will appreciate (Veres and Mansson 2005; Alexander and Stevens 2002). Mental models are thought in some way to resemble “real life” situations (Johnson-Laird 2005). But today’s conceptual models do not resemble the situations they describe; they are more like abstract statements of fact. In this regard they may be poorly matched with mental models. To minimize cognitive load, perhaps conceptual models should more closely resemble the situations they represent. Interpretation could then become largely unconscious and instantaneous (Burek 2005).

7.3. Principle 3: Reinforcement

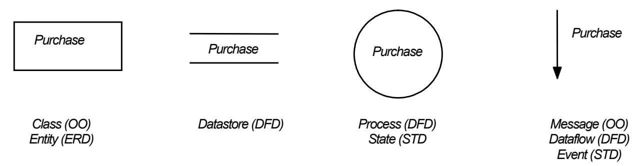
Maximize Comprehension by Combining Words with Pictures. Verbal and visual sensory inputs are processed independently but the two cognitive systems are thought to interact to improve recognition when both visual and verbal cues are present (Medin, Ross and Markman 2004). This is why combining images with words is superior to the use of words alone; an image acts as a powerful stimulus for association (Siau 2005). The images need not be photographic in nature since the mind fills in details automatically; simple drawings can easily stimulate recognition. Research into the visual processing of cartoons has shown that only a small portion of the information in an image is required for recognition. As in graphic design, visuals need only be rich enough to convey the necessary information (Goldstein 2005). When used well, icons and other pictorial representations exemplify this principle (Chen 1999).

7.4. Principle 4: Consistency

Consistently Use Each Symbol with Only One Business Meaning. Meaning is created when an observer receives sensory cues and recalls knowledge by association. For this to happen reliably, the cues must have well-defined and distinct associations. Therefore there must be consistency in the way that symbols are used; we must take care to link each symbol with only one concept and vice versa. In a conceptual model the requirement for consistency means that each distinct end user concept (such as *purchase, customer,* and *product)* should be represented using its own meaningful and distinct symbol. This idea goes against current practice; most conceptual modeling methods do not relate a unique symbol with each concept. The same concepts may be represented using different symbols at different times and any given symbol may mean several things (Shanks, Tansley and Weber 2004). To illustrate, **Error! Reference source not found.** shows symbols representing the concept *purchase*. Several of the symbols are used with multiple meanings. Each symbol represents some abstract aspect of the mental concept *purchase* rather than the concept itself.

The lack of a 1-1 correspondence between mental concept and symbol creates confusion and makes mental association impossible; therefore it is harder for a non-expert to understand the model. The potential for preattentive processing, with rapid comprehension and visual searching, is lost. Even experts may expend more cognitive effort than necessary in interpreting models.

Figure 7 Representations for Aspects of Concept *Purchase*



7.5. Principle 5: Chunking

Support Short-Term Memory with Visual Chunking Strategies. Our limited attention capacity makes it easy to become overloaded by information; according to cognitive load theory we can focus on only a few items simultaneously (Bannert 2002). We are therefore obliged to filter information or to translate it into a more easily-assimilated form. This is called *chunking*: to avoid attending to many concepts simultaneously, we place them into meaningful groups and then treat each group as a single concept (Gobet 2005).

Chunking naturally forms hierarchical relationships between concepts. Hierarchies improve comprehension and retention; they mimic the expert's organization of knowledge and assist recall (Goldstein 2005). Chunking could be described as the "divide and conquer" strategy: tackling a large problem by dividing it into sub-problems. One criticism is that this approach militates against a holistic view and can result in poorly integrated, partial solutions. Therefore, it is important to have a way of retaining the "big picture" when chunking is used.

In conceptual modeling, diagrams routinely contain hundreds of items and information overload is common. It is difficult to deal with large diagrams, especially when associations are represented as lines, which can make it hard to place items optimally. Techniques for hiding complexity are needed (Moody 2004). Table 15 summarizes the rather limited support in some standard modeling techniques for the main chunking strategies *summarizing, filtering* and *partitioning*, suggesting that such support for chunking is patchy. For instance, no technique offers filtering strategies; one cannot easily simplify a class diagram by hiding all classes except those representing people and organizations, unless suitable inheritance hierarchies happen to be present. Arguably, no current conceptual modeling technique permits easy and flexible generalization about situations and processes as suggested, for example, by the theories of situational memory and scripts (Schank 1999).

Table 15 Strategies for Reducing Complexity

Technique	Summarizing	Filtering	Partitioning
Object modeling (e.g. UML)	Inheritance	None	Aggregation; Packages & subset diagrams (informal)
Data modeling (e.g. E-R)	Supertypes	None	Subject areas (informal)
Process modeling (e.g. DFD)	None	None	Decomposition
Behavior modeling (e.g. STD)	Nested states	None	None

In practice, models are divided into arbitrary subsets, typically by drawing partial diagrams on separate pages. The subsets are not necessarily meaningful in themselves. Managing the parallel evolution and reintegration of separate and potentially conflicting model subsets is a challenge (Ramesh and Dennis 2002). It can be difficult to retain an overview. Object modeling offers the formal grouping construct *aggregation,* which may better reflect real-world relationships. However, like all of the currently-available chunking strategies, aggregation is a *static* construct; it affects model structure and cannot be applied dynamically as required. Hence the modeler is limited at present in the choice of chunking strategies and must make structural decisions to use them. This is not ideal, since structures like inheritance and aggregation have implications for the further development of models and the systems designed using them. There may be benefits in visual chunking strategies that can be employed flexibly and without the need for structural change to a model.

7.6. Principle 6: Fuzzy Categories

Support Alternative Concept Definitions and Concept Instability. Mental categories are inherently fuzzy. We may know what we mean by a category but find it difficult to state the criteria we use to categorize (Eysenck and Keane 2005). Some things seem more typical of their categories than others and some properties more important than others in governing category membership (Burek 2005; Solso, MacLin and MacLin 2004). Categories may be defined intensionally (e.g. *tricycle: a three-wheeled vehicle propelled by pedaling*) or extensionally, by enumerating the members of the group, or in terms of prototype or exemplar objects (Medin, Ross and Markman 2004). It seems that we unconsciously apply a combination of these methods. Context is also important: the tendency to classify things in different ways according to context is termed *concept instability*.

The connectionist view of brain function suggests that categories are not a primary mechanism for cognition. Instead, we seem to use a form of pattern matching; one thing reminds us of another. The combined effect of interacting neurons results in this behavior without any need for mental categories or conceptual structures. Like the human mind, artificial neural networks also categorize effectively on the basis of exposure to known patterns. Like humans they learn by experience and can generalize their knowledge to new areas, and they do this without forming rules or conceptual structures. According to this view, categories are emergent, conscious, verbal phenomena, and come into being only when we consciously try to formulate them or attempt to define a term. The bulk of mental activity is unconscious, yet our understanding of categories may be modified by conscious reasoning; for example, a dolphin may intuitively seem to be a fish but we may know consciously that it is a mammal.

All of this is problematic for conceptual modelers, who operate from the ontological position that concepts are well-defined ("crisp") and, in some sense, pre-existent. The inflexibility of our conceptual modeling techniques makes it difficult to model mental concepts satisfactorily. We must fit fluid ideas into static conceptual models and even invent concepts to satisfy the rigors of the technique (the introduction of "intersection entities" to replace many-to-many relationships in a data model is a classic example of this). It might be more useful if modeling techniques worked more like the brain, perhaps allowing concepts to be defined using multiple strategies or even to remain undefined, at least for some of the time.

7.7. Principle 7: Brainstorming

Capture Unstructured Ideas and Allow Easy Model Restructuring. It is often difficult for groups to reach consensus; research shows that effective groups seek compromises rather than optimal solutions. Groups are more productive when criticism and competition are reduced, and participants are more likely to contribute if the work is aimed at exploration and learning rather than unanimity. In brainstorming, an absence of judgment and editing is known to encourage group creativity (Zander 1994).

A similar situation occurs in conceptual modeling; especially during the earlier stages, when ideas are thrown around and experimentation with alternatives is often needed. Substantial revision is normally necessary. Yet conceptual modeling tools typically require ideas to be expressed formally at the time they are recorded. We have to model in correct syntax; ill-formed class definitions cannot be recorded. To work informally one must abandon the model and resort to flip chart and whiteboard. Once a model has been started it can be difficult to change tack and to restructure it along new lines; there is a penalty for not getting it right first time. Arguably, this need for early rigor is counterproductive when used in a group setting, since it prevents the very tolerance and lack of editing that groups need to function well. Perhaps conceptual modeling techniques and tools should tolerate incompleteness, incorrectness and inconsistency. Modelers should be able to refine models at their own pace until they reach a satisfactory state. They should be able to attend to the problem at hand and not be forced to resolve inconsistencies and fix syntax at any particular stage.

7.8. Principle 8: Error Tolerance

Tolerate and Reduce the Likelihood of Simple Errors. It goes without saying that modelers are imperfect and rarely get it right first time; the modeling process is inherently iterative. Attention cannot be sustained indefinitely and modelers inevitably make mistakes (Leung and Bolloju 2005). This is not too problematic for expert modelers, who find it relatively easy to identify and resolve errors. But it is much more difficult for non-experts to correct mistakes, especially when the mistakes are not obvious. Conventional conceptual modeling techniques offer plenty of opportunity for error; for instance, techniques which incorporate built-in redundancy create the possibility of inconsistency. One example occurs in data modeling, where the association of two entity types is equivalent to the presence of a foreign key attribute; novice modelers often fail to recognize this equivalence and define attributes that contradict associations. For the expert, redundancy may be a boon, since it can highlight inconsistency and thereby reveal inadequate analysis; double-entry accounting is based on this principle. But for non-experts redundancy is typically confusing and often leads to inconsistency which remains unresolved.

Other common problems centre on naming and terminology: model elements with distinct meanings may be named identically and the same facts may be encoded several times in different ways. Lacking a clear mental framework for modeling, the novice is unlikely to spot errors of this type without external assistance (Rosenberg and Scott 2001). Yet current modeling techniques allow "semantic" errors like these to occur and leave it up to the modeler to identify them.

Greater opportunities for error and inconsistency exist when multiple modeling techniques are combined, as is common with UML and similar methods. For example, information in a use case diagram can contradict that in a class diagram. Often, the conflict is not obvious and recognizing it can require detective work together with the kind of insight that develops only with experience. Perhaps it might be better if conceptual modeling techniques were robust enough to ignore or tolerate errors like these. Ideally, the notation would make at least some errors impossible; one way of reducing the likelihood of error is to reduce the number of ways in which any given fact can be expressed. Another suggestion is to introduce semantic "intelligence" into the technique so that it can be self-checking to some extent; tool support can then bring errors to the modeler's attention and assist in their resolution.

7.9. Principle 9: Arousal and Attention

Use All Means Available to Maintain Attention. *Arousal* refers to the level of activity within an individual's brain, corresponding roughly to the individual's degree of alertness. A moderate level of arousal is necessary for effective comprehension; too little and too much arousal are both counter-productive. Arousal is increased by stimulating factors such as surprising or novel events, and by the use of certain drugs, movement, noise and bright lights. It is decreased by monotonous or boring tasks and repetitive stimuli.

Working groups typically need to maintain attention for sustained periods. But in a group setting, with diminished individual responsibility, it is easy to become bored and distracted; concentration (i.e. arousal) tends to decrease. Traditional conceptual modeling sessions are often sedate affairs that use static, monochrome diagrams, resulting almost inevitably in lowered attention levels. The modeler can address this issue, however. Attention can be recaptured, for example, by introducing colorful, animated model representations and novel events into modeling sessions. Increased personal involvement by group members and frequent, enforced changes in task and posture can also help ensure that arousal is stimulated.

Research has shown that personally relevant information is much more interesting than other information. The issue of personal relevance can be addressed by expressing subject matter in the user's own language rather than that of the modeler, and by using notations that seem meaningful to the user rather than to the modeler. This would point, for example, to the use of realistic images rather than neutral symbols to denote concepts, and to the use of meaningful categories such as "person", "document" and "place" rather generic terms such as than "class" and "property". These actions may also tend to create the right "set" in the mind of the user; this is addressed in our final principle, below.

7.10. Principle 10: Set

To Instill Helpful Sets, Make the Purpose and Method of Modeling Obvious. One of the most important factors affecting success in problem-solving is the range of characteristics known as *set*, a predisposition towards viewing a situation in one particular way. In conceptual modeling, framing influences can easily create sets which affect problem understanding and decision-making (Adams and Avison 2003). These sets limit the ability of both analyst and end user to view a problem situation holistically. One consequence is *loafing*; end-users often feel that models are 'technical' and the property of systems developers. The intention may be to capture relevant aspects of the user's world but participants view the models as more relevant to the analyst's work than to their own (Bansler and Bødker 1993).

This type of set has several origins. Users may be more interested in their day-to-day work and fail to understand the modeling process. Conceptual modeling may use unfamiliar notations and terms. The result can be confusion; analysts persist with inappropriate models, thinking mistakenly that they reflect the user's reality, while users "go along with" models they understand poorly. Few researchers have investigated the ability of end users to understand conceptual models (Topi and Ramesh 2002). However, it seems that these problems could be reduced by avoiding technical terms and abstractions in favor of the user's own concepts and language. Making models more intuitively understandable may allow non-experts to work with them and thereby develop a sense of ownership.

Another problem for conceptual modelers is *set of function*. This set hinders problem-solving: individuals fail to recognize the tools to resolve a situation if they must be used in unconventional ways. Experts often use ingenuity to model complex situations whereas novice modelers may not have sufficient knowledge to do this. Education can help, but in-depth training may be onerous and impractical (Dennis et al 1999). Perhaps the problem could be eased by making models more intuitively accessible so that the method of modeling is more obvious. As in user interface design, a well-designed model may be the one that needs least explanation (Nielsen and Loranger 2006).

Another way of helping to ensure that end users understand conceptual models might be to make end users responsible for their own models. This would certainly reduce the risk of social loafing (Dennis et al 2005). However, it would require modeling techniques to be no more difficult than, say, building spreadsheets. It would also require a change in set by IT professionals, who may see modeling as a technical process that they must regulate (Bansler and Bødker 1993). Modeling would have to be viewed instead as a user-controlled process in which the user's own concepts, terminology and agenda were paramount.

8. Appendix B: Business Concept Modeling (BCM) Technique

BCM is a modeling technique whose subject matter is similar to that of E-R diagrams and UML class diagrams. However, BCM is more business-oriented and this is reflected in the information content of models. BCM relies for model representation and manipulation on the use of a supporting software tool. One hallmark of BCM is its support for easy reuse of business concept definitions; this means that it is rarely necessary to construct a new model completely from scratch.

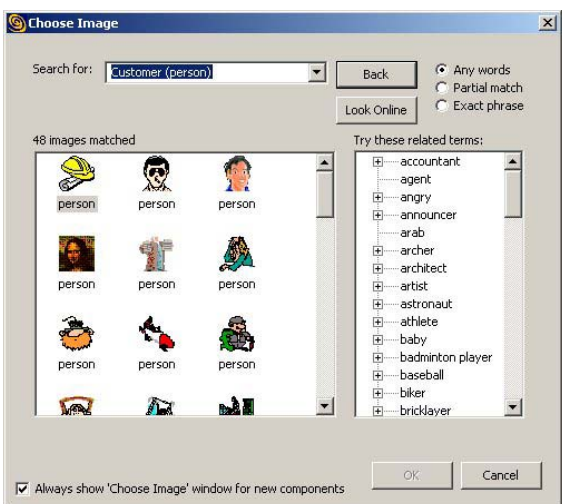
By virtue of the supporting tool, BCM honors the *error tolerance* and *isomorphism* principles (see Appendix A) using meaningful pictures to represent business concepts and associating each business concept with an "innate type" (Table 16). The innate types consist of nine predefined categories, corresponding to commonly-understood ideas that we might expect most people to have. The list of innate types is comparable with the "people, places, things and events" of object modeling (Bolloju 2004) and with upper ontologies based on philosophy or natural language (Everman 2005; Shanks, Tansley and Weber 2003).

Table 16 BCM Innate Concept Types

Concept type	Represents	Example
Person	Individual person	Claims adjuster
Organization	Identifiable group of people	Insurance company
Activity	Event, process or activity; something that happens	Purchase
Place	Physical location	Supermarket
Physical object	Concrete, physical object	Car
Document	Information on paper or in electronic form	Bank statement
Category	Way of grouping or classifying things	Gender
Idea	Conceptual object, information in abstract form	Law
System	Technology such as computer-based IS	Payroll system

BCM partially honors the *fuzzy categorization* principle by allowing concepts to be freely associated and re-associated with different types and with each other. This flexibility in concept definition also helps fulfill the *brainstorming* principle. In addition, color, texture and task variety are used to retain interest in accordance with the *arousal and attention* principle.

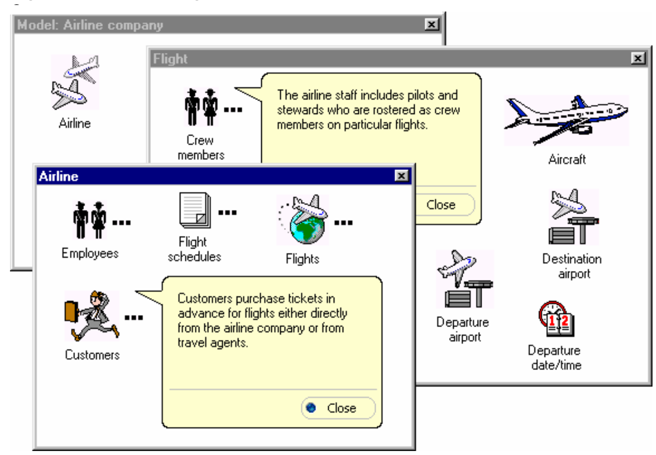
Figure 8 Choosing an Image



To meet the *consistency* principle, BCM allows each business concept in a model to be represented using a unique image. The modeler selects images from alternatives suggested by the tool (Figure 7). The images are semantically linked (thesaurus-style) so that the modeler can browse through associated terms to find potentially suitable images. This feature is especially useful since some concepts are best expressed through visual metaphors; semantic links allow users to find tangentially-related images which can often be used with a metaphorical relationship to the concept of interest.

The requirement to use a distinct image for each concept is an important one: apart from capitalizing on the power of preattentive processing it helps to meet

Figure 9 Use of Images and Text to Represent

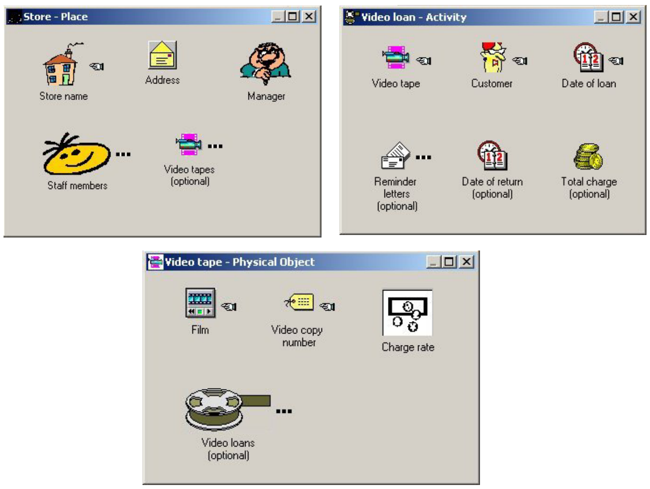


the *reinforcement* principle, especially when supporting text is used (Figure 7). The images can include icons as well as any other graphics that the user would like to use.

Relationships are added to a model by placing images in windows; each window represents a distinct concept (*visual chunking* principle). Relationships between concepts are defined by the presence of images in windows rather than by lines or arrows. This capitalizes on the commonly-understood "containment" relationship between windows and icons; it is intended to make model structure more intuitively understandable.

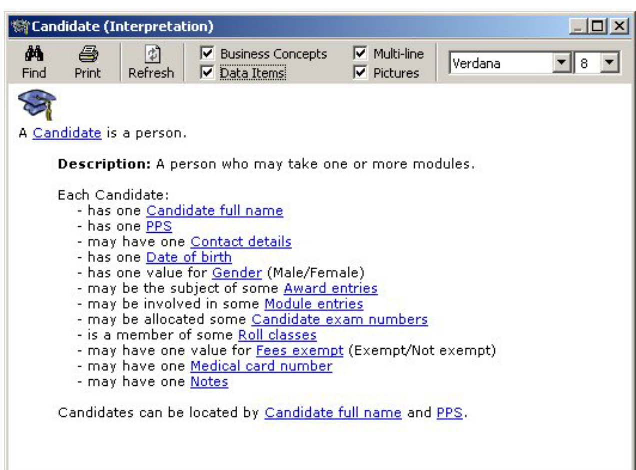
The example in Figure 9 illustrates relationships between three concepts: *store, video tape* and *video loan*; the three dots next to a concept indicate *many* whilst the absence of dots indicates *one*. The pointing hand indicates concepts that are most typically used to locate or select instances of the corresponding window's concept (for example, a store can be located by store name).

Figure 10 Relationships Between Concepts



Overall, BCM is designed to avoid the culture shock that many users experience when confronted by "box and arrow" notations. The images and familiar window-icon interface help fulfill the *set* principle. An English-language interpretation (Figure 10) provides an alternative, non-technical, way of understanding the model.

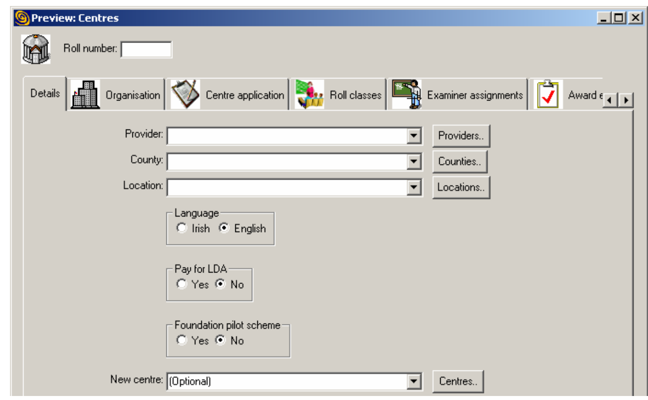
Figure 11 English-Language Interpretation



Additionally, a form preview feature (Figure 11) lets users visualize the result of their model in the form of prototype form designs. The form preview feature applies heuristic algorithms to generate reasonably usable form layouts.

For more advanced users, the BCM tool incorporates a hierarchical concept explorer, a model checking facility and a UML-style class diagram view. The tool also allows models to be converted into working software applications automatically; the user chooses which platforms they want to run the application on. The tool currently supports Microsoft Access databases and forms, Oracle databases and Java/J2EE web applications; Microsoft ASP and SQL Server have also previously been supported.

Figure 12 Form Preview



9. Appendix C: Experimental Procedure

The goal was to observe modelers in action and to analyze their performance as they developed models. To this end a number of modeling sessions were conducted; each model was produced using either BCM or object modeling. Quantitative data was captured about each model and interpreted in the light of qualitative data gathered during the modeling sessions using participant observation, questionnaires and interviews.

9.1. Participants

Ten of the models were produced by modelers in sessions with groups of business end users (Justice and Jamieson 2006); group members were chosen according to the relevance of their business knowledge and experience. Nine models were produced by individual modelers working alone, using their own business knowledge or consulting with users in interviews. All modelers were screened in advance for prior knowledge of object or data modeling. They ranged from business people with little IT knowledge to an experienced data analyst who was expert in object and data modeling (Table 6). None of the modelers had any prior knowledge of BCM. The possibility of knowledge transfer between the two techniques was avoided by exposing each participant to only one technique (except for the expert modeler, who used both).

9.2. Method

Each modeler constructed a model from scratch, revising it in subsequent sessions until it was considered complete and correct. This process resulted in a series of versions for each of the nineteen models. Every version was numbered and dated; subject area, session time, venue and participants were recorded for each group session. When each modeler deemed a model to be complete it was inspected and any deficiencies noted. A final "corrected" version of the model was produced by making the least disruptive changes required to bring the model into a correct state. This final version represented the baseline against which prior versions of the model would be compared. The correction process was repeated in an effort to help ensure an unbiased result.

Object modeling was supported for group sessions by whiteboard and overhead projector, and object models were rendered using the System Architect CASE tool outside of modeling sessions. BCM was supported by the specially-designed software tool in modeling sessions.

Each modeler received a 60-minute practical introduction to the relevant modeling technique, followed by periodic reviews to assess progress and to assist where possible. To avoid uncertainty, the scope of each model was agreed in advance; modelers were told that the goal was to state data requirements for a database which would support the specified business area. As a starting point modelers were asked to think about (or get group members to talk about) their work and organizations and to identify relevant concepts which would form the basis of their model. Questions about modeling technique were answered fully at all stages; however, modelers were not told specifically how to model their business areas. Since assistance was a potential source of bias, efforts were made to avoid giving greater assistance to one group of modelers over the other.

To help ensure a representative result, modelers were matched between groups (Table 17) and triangulation was used between the quantitative data (from model analysis) and the qualitative data in which participant observation and interpretation played a key part. A secondary study was conducted for comparison. Care was taken to provide a fair and rigorous comparison between the two modeling techniques; preconceived ideas were 'bracketed' (LeVasseur 2003). Reliability could also have been affected by the number of participants in each modeling session, by the choice of participants, and by the time spent on each modeling session. Efforts were made to keep these factors constant between the two modeling techniques.

Table 17 Distribution of Models by Experience Level and Technique

Technique	Number of models		
	No experience	Little experience	Expert
Object Modeling	2	3	5
BCM	3	2	4

9.3. Quantitative Measures

Each model version was analyzed to determine the raw measures listed in Table 4. To help ensure reliability, models were coded electronically; this allowed them to be analyzed and compared automatically by a specially-designed tool. The results were plotted graphically and this helped emergent patterns and correlations to be observed. It also permitted exploratory analysis; for example, graphs showing emergent relationships between model size and change rates were produced only after a number of alternative analyses were done.

9.4. Qualitative Measures

Qualitative results came from three main sources: questionnaires completed by modelers and group members, notes from interviews with modelers and observations by the researcher. Each participant was asked to complete an initial questionnaire; the aim was to determine attitudes and assumptions as well as prior knowledge of relevant areas such as business and requirements analysis. After each session, participants commented in a further questionnaire about their impressions and understanding of the techniques used. They rated the completeness and correctness of the models. The purpose was to establish a snapshot of participants' thinking against which later results could be compared. In particular, this was intended to allow subjective assessments of model quality to be compared with more objective measures from the analysis of models. Notes taken by the expert modeler also provided a qualitative record of the proceedings in group sessions. The modeler assessed each participant's contribution as well as their specific areas of difficulty or clarity.

10. Appendix D: Questionnaire Responses from Group Members

Analysis of responses was hampered by a rather low number of returned questionnaires. Overall, there seemed few qualitative differences between the reported experiences of group members who used either method; it is clear that participants generally understood the process they were part of.

Respondents who had created a database before were more likely to judge their models as complete and correct. The value of the sessions was rated more highly by those who had created databases before and those who were educated to diploma or degree level. Those who participated in a second (i.e. follow-up) session were more likely to judge the model as complete and correct.

57% of those with experience of creating a database were able to correctly state the purpose of the sessions whereas only 25% of those without experience could identify the purpose of the modeling sessions. Participants who had never created a database were more likely to state the purpose of the sessions as modeling processes rather than data, suggesting that creating a database provides insight into the reasons for modeling. Respondents who correctly identified the purpose of the modeling session were more likely to judge the models as complete and correct and to rate the sessions as valuable.

1