Information Modeling and Method Engineering: A Psychological Perspective

Keng Siau
University of Nebraska-Lincoln, USA

Information modeling is the cornerstone of information systems analysis and design. Information models, the products of information modeling, not only provide the abstractions required to facilitate communication between the analysts and end users, but they also provide a formal basis for developing tools and techniques used in information systems development. The process of designing, constructing, and adapting information modeling methods for information systems development is known as method engineering. Despite the pivotal role of modeling methods in successful information systems development, most modeling methods are designed based on common sense and intuition of the method designers with little or no theoretical foundation or empirical evidence. Systematic scientific approach is missing! This paper proposes the use of cognitive psychology as a reference discipline for information modeling and method engineering. Theories in cognitive psychology are reviewed in this paper and their application to information modeling and method engineering are also discussed.

Even though research in systems analysis and design has been going on for over 40 years, successful software development is still an art rather than a science. In the 1980s, Jones (1986) observed that a typical project was one year late and 100% over budget. Yourdon (1989) reported application backlogs of four to seven years or more. The maintenance phase typically consumed up to 70% of the programmer’s effort, and it was errors, not enhancements, that accounted for 40% of maintenance (Rush, 1985). Page-Jones (1988) wrote: “It looks as if traditionally we spend about half of our time making mistakes and the other half of our time fixing them.”

We are, however, no better as we move toward the end of this century. The IBM’s Consulting Group (Gibbs 1994) released the results of a survey of 24 leading companies that had developed large distributed systems. The numbers were unsettling: 55% of the projects cost more than budgeted, 68% overran their schedules, and 88% had to be substantially redesigned. A recent high-profile failure is the Denver Airport baggage-handling system, responsible for delaying the opening of the airport. The Standish Group research (Chaos 1995) predicted that a staggering 31.1% of projects would be canceled before they ever get completed and 52.7% of projects would cost 189% of their original estimates.

In the early days of computerized information systems, technological failure was the main cause in the failure of business data processing systems (Avison & Fitzgerald 1995). Today, the failure of information systems is rarely due to technology that is on the whole reliable and well tested. Failure is more likely to be caused by miscommunication and misspecification of requirements. Similar sentiments were echoed in the Standish Group’s report (Chaos, 1995) which listed incomplete requirements and specifications as the second most important factor that caused projects to be challenged and the top factor that caused projects to be impaired and ultimately canceled (Chaos, 1995). A recent survey of hundreds of Digital’s staff and an analysis of the corporate planning database revealed that on average, 40% of the requirements specified in the feasibility and requirements phase of the life
cycle were redefined in the later phases. This cost Digital an average of 50% more than the budgeted amount (Hutchings & Knox, 1995).

The process of investigating the problems and requirements of the user community, and building an accurate and correct requirement specification for the desired system is known as information modeling (Siau, 1999; Siau & Rossi, 1998; Siau et al., 1997; Mylopoulos, 1992, Rolland & Cauvet, 1992; Kangassalo, 1990).

Information Modeling

Information modeling is the process of formally documenting the problem domain for the purpose of understanding and communication among the stakeholders (Siau, 1999; Siau, 1998; Mylopoulos, 1992). Information modeling is central to information systems analysis and design, and takes place in the early phases of the software development life cycle. The product of the information modeling process is one or more information models (e.g., data flow diagrams, entity-relationship diagrams, use cases, activity diagrams, sequence diagrams). Information model provides a conceptual basis for communicating and thinking about information systems (Willumsen, 1993), and a formal basis for tools and techniques used in the design and development of information systems (Kung & Solvberg, 1986).

Information models are constructed using information modeling method, which can be defined as an approach to perform modeling, based on a specific way of thinking, consisting of directions and rules, and structured in a systematic way (Brinkkemper 1996). There is no shortage of information modeling methods in the field. In fact, it is a “methodology jungle” out there (Avison & Fitzgerald, 1995). Olle et al. (1982) and Bubenko (1986) stated that the field was inundated by hundreds of different modeling methods. Recently, Jayaratna (1994) estimated that there were more than a thousand brand name methodologies worldwide. The quest to develop the next modeling method has been wittily termed the YAMA (Yet Another Modeling Approach) syndrome (Oei et al., 1992) and NAMA (Not Another Modeling Approach) hysteria (Siau et al., 1996). Even the new kid on the block, object oriented approach, has more than a dozen variants. Despite the “impressive” number, miscommunication and misspecification continue (Chaos, 1995).

To reduce the chances of misunderstanding and miscommunication during information modeling, the use of natural and intuitive modeling constructs (e.g., entity, relationship, object) in information modeling methods has been stressed and advocated (e.g., Chen, 1976; Coad & Yourdon, 1991). This, they claimed, would enable end-users to better understand the information depicted in the information model and to pinpoint incomplete or incorrect information in the model.

Method Engineering and Modeling Constructs

Modeling constructs are semantic primitives that are used to organize and represent knowledge about the domain of interest (Sernades et al., 1989). Modeling constructs form the core of an information modeling method. Method engineering is the process of designing, constructing, and adapting modeling methods for the development of information systems (Siau, 1999; Siau, 1998; Brinkkemper, 1996). To design, construct, and adapt methods, we need to understand the role and value of each modeling construct.

The importance of modeling constructs can be viewed from two perspectives: ontology and epistemology of information systems analysis and design. Ontology is concerned with the essence of things and the nature of the world (Wand & Weber, 1993; Avison & Fitzgerald, 1995). The nominalist position in ontology argues that “reality is not a given immutable ‘out there’, but is socially constructed. It is the product of human mind” (Hirschheim & Klein, 1989).

The choice of modeling constructs, therefore, directly influences what the modeling method regards as important and meaningful versus what it suggests as unimportant and irrelevant. For example, the use of the entity-relationship(ER) approach emphasizes entities and relationships but ignores the processes involved. The use of the object-oriented (OO) approach, on the other hand, emphasizes objects and the behavior of objects.

Epistemology relates to the way in which the world may be legitimately investigated and what may be considered as knowledge (Avison & Fitzgerald, 1995). The choice of modeling constructs constrains how one can know or learn about reality—the basis of one’s claim to knowledge (Klein & Lyytinen, 1983; Walsham, 1993). Users of the entity-relationship approach, for example, would focus on identifying entities and relationships whereas users of data-flow diagram (DFD) would emphasize the eliciting of processes, data flows, external entities, and data stores from the problem domain.

Despite the importance of modeling constructs, not much research has been done in this area. Most modeling constructs are introduced based on common sense, superficial observation, and intuition of researchers and practitioners. Theoretical foundation and empirical evidence are either non-existent or considered non-essential. For example, Coad and Yourdon (1991, p. 16) nicely summed up the practitioners’ scant concern:

“It would be intellectually satisfying to the authors if we could report that we studied the philosophical ideas behind methods of organization, from Socrates and Aristotle to Descartes and Kant. Then, based on the underlying methods human beings use, we could propose the basic constructs essential to an analysis method. But in truth we cannot say that, nor did we do it.” (emphasis added)
Related Content

Integrity Maintenance In Extensible Databases
Ulrich Schiel (2002). Database Integrity: Challenges and Solutions (pp. 322-334).
www.igi-global.com/chapter/integrity-maintenance-extensible-databases/7886?camid=4v1a

Data Warehousing Interoperability for the Extended Enterprise
Aristides Triantafillakis, Panagiotis Kanellis and Drakoulis Martakos (2004). Journal of Database Management (pp. 73-84).
www.igi-global.com/article/data-warehousing-interoperability-extended-enterprise/3317?camid=4v1a

Unifying Access to Heterogeneous Document Databases through Contextual Metadata
www.igi-global.com/chapter/unifying-access-heterogeneous-document-databases/9207?camid=4v1a

BROOD: Business Rules-Driven Object Oriented Design
www.igi-global.com/chapter/brood-business-rules-driven-object/39349?camid=4v1a