Information systems are becoming ubiquitous in our lives. They are used to pay our salaries, they co-ordinate the activities of international companies, and they link the world together in global networks. The purpose of an information system is to support communication and work within and between organisations. An information system must provide accurate and up-to-date information that satisfies the information needs of enterprises, thereby supporting their routine operations as well as their decision making. In their most basic forms, information systems process and store large amounts of data from routine business transactions, such as payrolls, invoicing, and stocktaking. They support the day-to-day activities of a business by relieving people from the tedious and time-consuming aspects of performing operational transactions. An information system can also provide managers and decision makers with information about the activities of their organisation, thereby helping them in spotting business opportunities, detecting long-term trends, and monitoring the performance of the organisation. An information system can also include more advanced tools for analysis and decision making, e.g. for optimisation and data mining. In order to take strategic decisions, external information about the environment of an organisation is at least as important as internal information. Information systems, therefore, also have the role of supporting management with such external information, known as business intelligence.

Information systems belong to the most complex artifacts built in today’s society. Developing, using, and maintaining an information system raises a large number of difficult problems, ranging from purely technical to organizational and social ones. Many of these problems are ill-structured, meaning that there are no algorithms or mechanical methods for solving them, or that they cannot even be precisely formulated. The problems are ill-
structured mainly because the development and use of an information system involve many kinds of stakeholders with different and conflicting interests and perspectives, which need to be sorted out and negotiated. This is a difficult task as information systems and services are notoriously hard to illustrate and describe in terms that are easily understandable to non-experts. Communication problems are rather the rule than the exception. There is no panacea for these problems, but there are aids by which the problems can be described more clearly, in a more structured way, and sometimes even be formally represented. These aids have been investigated within the area of information systems engineering. They consist of solid conceptual frameworks and clear notations to be used when describing and designing systems at the conceptualization and problem formulation level. Such frameworks and notation as well as associated methods, called enterprise modeling, can significantly improve the dialogue and cooperation between stakeholders in information systems design and use.

An enterprise model is a "computational representation of the structure, activities, processes, information, resources, people, behavior, goals, and constraints of a business, government, or other enterprises", as defined in (Enterprise Modeling, 2007). Enterprise models have been used in information systems design, and it is possible to identify three main ways of utilising enterprise models, (Fowler, 2003):

- **Models as sketches.** Models are used as sketches to describe possible solutions to problems or to document existing solutions in order to facilitate communication among stakeholders. The idea is to use the models as informal support for communication and description.

- **Models as blueprints.** Models are used as blueprints for implementing information systems and services. The idea is that the models shall be sufficiently precise and formal for programmers, database designers and other IT experts to build a functioning system.

- **Executable models.** Executable models take the idea of models as blueprints one step further. The models shall be formal enough to be automatically translatable into executable code. In this way, the coding step is eliminated, thereby reducing cost and risk for introducing errors.

Enterprise models have been used in these ways for a long time in business and systems design, but they have not yet been put to their full potential. Typically, they have been used only as sketches for limited tasks in systems design and then discarded. To realise the full potential of enterprise models, there is a need to use models as blueprints and taking advantage of
executable models. This can be realised by a business and technology architecture that places the models firmly in the centre and let them be the driving force in analysis, design, implementation, deployment and use of systems and services. There is a need for a business and model driven information systems architecture. Such an architecture will serve to enact the software specification contained in the models by composing the software executable at the time of need. This will remove the lag between changes in the model and changes in the software system driven by that model, and will create software systems that evolve seamlessly to adapt to the rapidly changing business and organisational practices. Dynamic composition will enable the delivery of fine-grained software as services personalised to the user needs at the point of delivery. This will create unique working environments tailored to, and controlled by people. A business and model driven information systems architecture will thus put people at the centre of new working environments supported by model-driven software services.

Much research is still needed before a business and model driven information systems architecture can be realised. A number of key issues in current enterprise modelling research are the following:

What is the right balance between expressiveness and usability in enterprise modelling languages? A language for enterprise modelling can be highly expressive, allowing for preciseness and reasoning support, often through some logic based formalism. This is fine for building advanced and comprehensive models, but the drawback is that the modelling process becomes more difficult, in particular for business experts with limited time and experience of modelling. A closely related question is how to improve the usability of enterprise modeling languages through adequate graphical notations.

- Which are the right concepts for modelling enterprises? There exist many different types of enterprise models: information models, conceptual models, activity models, process models, role models, goal models, business models, and so on. For each type of enterprise model, there also exist many alternative languages and notations based on different concepts. There is an ongoing search for the most appropriate concepts for capturing all the different aspects of enterprises, and how the resulting models are to be related to each other. While early enterprise models focused on data and information analysis, we see today a move towards modelling processes and value exchanges in networks of organisations.

- How should enterprise models be developed? Building enterprise models is a complex undertaking as it requires the contribution from many...
stakeholders with different perspectives, ranging from end users and business experts to management and information systems designers. Therefore, methods for building enterprise models are needed, and there is today more and more consensus that these methods need to be agile as well as participatory.

- How are enterprise models to be managed? As enterprise models become more and more wide-spread, there is a need to manage their relationships, maintenance, and evolution. Enterprise models are to be integrated, harmonised, made interoperable with each other, and related to standards. Thus, there is a need for methods, techniques, and tools that can help in these efforts.

This book identifies current trends as well as emerging and future areas in enterprise modeling and information systems engineering. The book also presents recent research results and experiences from applications in industry. The chapters in the book cover all of the issues introduced above ranging from analysis of data models over methods for participative modeling to the design of process and value networks.

The chapter entitled "Translating Schemas Between Data Modelling Languages" by Peter McBrien addresses the classical but still essential issues of data analysis and mapping. The chapter focuses on data modelling languages, and the challenges faced in mapping schemas in one data modelling language into another data modelling language. The chapter reviews the ER, relational and UML modelling languages (the later being representative of object oriented programming languages), highlighting aspects of each modelling language that are not representable in the others. The chapter introduces a nested hypergraph data model that may be used as an underlying representation of data models, which shows the differences between the modelling languages in a more precise manner. Finally, the chapter proposes a platform for the future building of an automated procedure for translating schemas from one modelling language to another.

The chapter entitled “Intention Driven Conceptual Modelling” by Colette Rolland discusses how classic conceptual modelling notions need to be complemented with intention and strategy driven modelling. The chapter argues that while conceptual models succeeded in telling us how to represent some excerpt of the world in informational terms, they failed to guide system analysts in conceptualising purposeful systems, i.e. systems that meet the expectations of their users. The chapter investigates the issue of conceptualising purposeful systems and discusses the role that goal driven approaches can play to resolve it. It considers the challenge of new systems having a
multifaceted purpose and shows how intention/strategy maps help facing this challenge.

The chapter entitled “Integrated Goal, Data and Process Modeling: From TEMPORA to Model-Generated Work-Places” by John Krogstie also addresses the issue of extending and combining modelling approaches including goals, data, and processes. Goals and rules on different levels ranging from visions, to strategies, tactics, and operational goals have been acknowledged for a long time. In the information systems engineering field, the interest in goals and rules has come from two directions, requirements engineering and rule based systems. Based on perspectives from these areas, the chapter proposes an approach to combining goal, data, resource and process modeling, in the support of the development and user-led evolution of what is called Model-generated Work-places.

The chapter entitled “Value and Intention Based Information Systems Engineering” by Paul Johannesson and Prasad Jayaweera investigates the problem of relating and structuring enterprise models of different types. The authors propose a light-weight enterprise architecture framework based on linguistic theories and organizational metaphors and argue that it provides a number of advantages in terms of flexibility, traceability and business orientation. The authors show how basic notions in enterprise modelling can be organised by three organisational metaphors - the organisation as machine where the focus is on the production and exchange of resources, the organisation as negotiated order where the focus is on commitments and contracts, and the organisation as power structure where the focus is on authorities and roles.

The chapter entitled “Pragmatic-Driven Approach for Service-Oriented Analysis and Design” by Remigijus Gustas and Prima Gustienė investigates novel modelling concepts for managing services and presents a pragmatic-driven approach for service-oriented information systems analysis and design. Services are viewed as dynamic subsystems, where outputs depend not only on inputs, but on a service state as well. Intentions of various business process experts are represented in terms of a set of pragmatic dependencies, which are driving the overall system engineering process. It is demonstrated how the pragmatic aspects are linked to the conceptual representations, which define the semantics of business design. In contrast to the traditional system development methodologies, the main difference of the service-oriented approach is that it integrates the static and dynamic aspects into one type of diagram.

The chapter entitled “The Practice of Participatory Enterprise Modelling – a Competency Perspective” by Anne Persson addresses the question of how to develop enterprise models in a participatory way, in particular which competencies that are required for this purpose. The chapter presents the two
main ways of working when it comes to involving stakeholders in the modelling process, the participatory and the non-participatory, and then focuses on the participatory approach. The author describes the desired competencies of domain experts and method experts, which are two of the most crucial actors in the participatory modelling process. The author further argues that although competency is one of the most critical success factors in modelling it is an overlooked topic in modelling research. The chapter is illustrated with quotes from an interview study that the author has carried out.

The chapter entitled “How to Support Agile Development Projects with Enterprise Modelling” by Janis Stirna and Marite Kirikova also addresses the issue of how to design enterprise models and discusses the role of enterprise models in the context of agile systems development, which has much in common with participatory design approaches. The chapter analyses the potential of using enterprise modelling in agile information system development projects on the basis of a number of empirical findings. The authors outline the current issues and challenges that projects using agile development approaches are facing. To address these challenges, they analyse what are the objectives of using enterprise modelling in agile development projects and give recommendations concerning modelling process as well as tool support.

The chapter entitled “Experiences with Modelling Early Requirements” by Pericles Loucopoulos investigates the early stages of systems development when requirements are to be identified and agreed upon. The chapter argues that a key challenge in the development of systems is the engagement of domain experts in their articulation, agreement, and validation of requirements. This challenge is particularly pronounced at the early requirements phase when multiple stakeholders from different divisions and often different organisations need to reach agreement about the intended systems. Decisions taken at this stage have a profound effect on the technical and economic feasibility of any project. The chapter introduces an approach that advocates the use of a modelling process expressed in terms of strategy-service-support dimensions, augmented by appropriate simulation techniques that enable experimentation with different scenarios. The chapter provides insights from a large project, in which the author played an active and interventionist part, on the utility of the approach in facilitating stakeholder engagement in early requirements specification.

The chapter entitled “Determining Requirements for Management Support Systems” by Sven Carlsson also addresses the management of requirements, but here focused on management support systems (MSS). The chapter presents a methodology that can be used as a guide for MSS design, with a primary focus on MSS requirements determination and how requirements can be fulfilled using information and communication technologies. The
methodology builds on Quinn and associates’ competing values model of organizational effectiveness and current MSS knowledge. The methodology can guide MSS designers in designing MSSs that support different managerial roles, i.e. the development of MSSs that support managerial cognition and behavior.

The chapter entitled “Towards a Holistic Approach to Validating Conceptual Models” by Jörg Becker, Björn Niehaves, and Daniel Pfeiffer moves into the issue of how to validate conceptual models. The chapter introduces a holistic approach to semantically validating conceptual models. The quality and thus the validation of conceptual models are of high economic importance, but only little empirical work has focused on their evaluation so far. This raises the question how a holistic approach for determining the quality of conceptual models should be designed. In order to describe the current state of research the authors develop a two dimensional framework and use it to identify a notable shortcoming on conceptual model evaluation, the lack of an approach that covers all aspects of the framework. To remedy this situation, the authors propose a procedure model that integrates different evaluation techniques, which provides a starting point to further elaborate on a holistic evaluation approach.

The chapter entitled “New Software Methodologies and Techniques for Business Models with Evolutionary Aspects” by Hamid Fujita proposes a language and an architecture for flexible information systems with a focus on eliciting and representing user intentions and requirements. The chapter also introduces a methodology, Lyee, for managing a development process moving from fuzzy intentions to formal and executable specifications. The results presented in the chapter are based on a large international project, which has resulted in numerous applications including the management of legacy software and the diagnosis of programs in imperative languages. The chapter gives an overview of the project and shows how its results have been developed and applied in industrial practice.

The chapter entitled “Refining the Concept Syndicate Data – Categories and Characteristics, Definitions, and a View Ahead” by Mattias Strand focuses on data management and introduces the concepts of external data and syndicate data. It contributes a conceptual discussion regarding different categories of syndicate data, as well as definitions and applications thereof. In addition, the chapter gives a view ahead for syndicate data, with respect to organizational as well as technological challenges and trends. Thereby, it increases the understanding of syndicate data as a vital component in business intelligence initiatives and explains why external data in general and syndicate data in particular have become prerequisites in modern information systems.

The chapter entitled “Interconnecting E-Business Model Components” by Eva Söderström and Vinay Kumar Mandala concerns electronic business (e-
business) models that have emerged and altered the traditional ways in which to do business. An e-business model is a specialisation of a business model, and is an organisational plan for how to work with the products or services intended to bring profit and revenue, and using ICT and the Internet in doing so. E-business models consist of components that constitute a business concept, i.e. an idea that an organisation wants to bring into reality. The analysis of the e-business model components and their interconnections results in a framework in the form of a model displaying the connections between the clustered components. The purpose of the chapter is to explain an overview of e-business model components and show how value is created and added at each component level.

The chapter entitled “Experiences from Technology Transfer Initiatives at SISU” by Janis Bubenko and Eva Lindencrona addresses the industrial uptake of information systems engineering and includes reflections on the authors’ experiences from an initiative to technology transfer in the field of information systems in Sweden. The authors discuss transfer of knowledge as well as of technical prototypes from academic research to product development, exploitation, and practical use in organisations. Their experiences emanate from their work in SISU – the Swedish Institute for Systems Development. The authors describe, firstly, which are the main products of technology transfer, and, secondly, which are the main factors that influence, or hinder, the success of a technology transfer initiative.

The chapter entitled “On IT-modelling in a cross-competence world” by Arne Sølvberg takes a broad perspective and discusses how IT in general and enterprise modelling in particular will become integrated into other disciplines. The author starts from the observation that the deep penetration of computers in all realms of society makes technological change the key driver for changing our lives. This will result in a change in our view of IT, from viewing the role of IT to mainly support other disciplines, to the integration of IT concepts, tools and theory into the modelling theories of the supported disciplines. The chapter discusses the opportunities and challenges of this integration.

This book is dedicated to Benkt Wangler, at his retirement from Högskolan Skövde, who has been a portal figure of the information systems area in Sweden for decades. Benkt Wangler is appreciated by his colleagues, fellow researchers and PhD students as a man of integrity, great knowledge on a wide variety of subjects, and with an ability to put complex problems into an understandable form. As a PhD supervisor, Benkt Wangler is dedicated and caring, for the actual research as well as for the students. As a colleague in the academic workplace, Benkt Wangler is known for his commitment to a wide variety of issues, both in the research group, the community of research leaders, the departments, and the university as a whole. Furthermore, his
knowledge and interest in art, literature and gourmet food has inspired many interesting conversations and excursions. Although not always strictly research oriented, these conversations have contributed to the widened knowledge and interest of all those participating. One highly valuable quality Benkt Wangler possesses is the ability to take new pieces of knowledge and research results and integrate them with his already existing vast knowledge into a comprehensive knowledge web. The results are new ideas, new research streams, new projects and a living and always evolving research community.

Paul Johannesson
Eva Söderström

References


List of Reviewers

Beatrice Alenljung, University of Skövde
Birger Andersson, Stockholm University
Per Backlund, University of Skövde
Maria Bergholtz, Stockholm University
Ananda Edirisuriya, University of Colombo
Martin Henkel, Royal Institute of Technology
Tharaka Ilayperuma, University of Ruhuna
Björn Lundell, University of Skövde
Erik Perjons, Royal Institute of Technology
Anne Persson, University of Skövde
Jelena Zdravkovic, University of Gävle
Rose-Mharie Åhlfeldt, University of Skövde