

Marlon Dumas, Wil M. van der Aalst, Arthur H. ter Hofstede, Process-Aware Information Systems: Bridging People and Software Through Process Technology, September 2005, Hardcover, 432 pp., ISBN: 0-471-66306-9, US \$99.95

This book claims that process technology is at the heart of an ongoing “silent revolution” that reaches back to the early 1970s and penetrates information systems until today. Many people consider process technologies—with workflow systems as one prominent technology instance—as fairly unsuccessful. However, one could also say that those technologies have been rather successful since they managed to be integrated in most information systems in use today and in that sense “disappeared” as a product market of their own but not as a concept and not as a set of technologies. This edited book is composed of four parts and contains 15 chapters in total. All authors are international experts in the domains they present in their chapters.

Part I (Concepts) consists of four chapters. After the introductory chapter which provides a detailed analysis as well as a historic perspective of process-aware information systems, a system for useful classification of process technologies is provided and discussed. Chapter 2 presents person to application processes. In particular, it discusses workflow management fundamentals which should be well understood before continuing with reading this book. Chapter 3 (written by Skip Ellis et al.) focuses on person to person processes and computer-supported collaborative work (CSCW). It is interesting to note that Skip Ellis is one of the first computer scientists working in the field of process-aware information systems and has been actively involved in this field since the early 1970s. What is striking about this chapter are the detailed characterizations of person to person interactions which are discussed as well as some recent software implementations representing those different interaction styles. Chapter 4 concludes the first part of this book and discusses Enterprise Application Integration (EAI). This chapter shifts the focus of attention to business to business interactions and discusses the required integration requirements as well as current architectures in the increasingly important field of EAI.

Part II (Modeling Languages) again comprises four chapters. Process modeling is fundamental for designing and implementing process-aware information systems. Chapter 5 outlines the current state of the art in using the UML for the purpose of modelling processes. It discusses in detail appropriate UML notations which can be utilized for such purpose, such as Activity-, Class-, Object-, Sequence-, and Structure diagrams including their strengths and weaknesses for the task at hand. Chapter 6 deals with process modeling using Event-driven process chains (EPC). EPCs are probably well known for those using SAP and ARIS systems. In that sense, this chapter can be understood as an introduction to EPCs and their underlying concepts. Chapter 7 discusses process modeling using Petri Nets. This formalism is well known and there is some literature on that theme. This chapter provides a concise and well-written introduction to Petri Nets. Chapter 8 is written by the editors of this book and discusses an emerging and increasingly relevant topic, namely, process modelling patterns. The authors categorize patterns which can be used for process modelling comprising various control-flow patterns. Similar to design patterns known in Software Engineering, the presented patterns may help in modeling processes and comparing process modeling systems. The main advantage of such patterns is that they are presented and discussed in a language independent way.

Part III (Techniques) consists of three chapters. Chapter 9 (Process Design and Redesign) deals with concepts not bound to technologies per se. The author discusses business process performance indicators (such as cost, quality, time, and flexibility) and presents best practices based on those indicators. This results in a useful overview of today’s indicators and techniques used to (re)design processes. Chapter 10 (process mining) presents the concept of extracting knowledge from event logs of information systems. The goal of this effort is to analyze underlying processes and to eventually improve future instances of processes (e.g., by performing delta-analyses between modeled and instantiated processes). The authors discuss examples of such logs and demonstrate clearly the benefits of a process mining approach. In particular this chapter presents the alpha-algorithm, which allows for mining the control-flow perspective of processes. Chapter 11 discusses transactional business processes which constitute an important backbone of today’s enterprise systems. Gustavo Alonso discusses fundamentals of transactions (e.g., ACID properties) before explaining the mapping to workflow processes and issues related to implementing technologies (e.g., Middleware).

Part IV (Standards and Tools) comprises four chapters. Chapter 12 presents standards for workflow definition and execution. This is achieved by presenting the leading standardization bodies and their goals, as well by presenting process definition languages as examples (e.g., XPDL, Wf-XML). Chapter 13 presents a more recent development: the Business Process Execution Language for Web Services (BPEL). After a short introduction to Web services the authors demonstrate the rationale behind creating a new language for Web service processes. This is achieved by providing small examples as well as a brief introduction to the syntax of BPEL. Important concepts known from traditional workflows including fault handling, compensation, and death path elimination are discussed as well. Chapter 14 discusses a particular workflow management system—Staffware (now part of TIBCO). This chapter (together with Chapter 15) is somewhat different from the rest of this book in the sense that it presents and discusses a particular product implementation. Chapter 15 presents FLOWer, a case-handling system. Before discussing the FLOWer system, the author also defines what is understood by the term case-handling and what the differences to traditional workflow systems are. Finally, the Appendix of this book provides a list of relevant additional resources, including books, web sites, standardization initiatives, and some tools.

The editors achieve their goals stated in the Introduction. This book provides a unifying and comprehensive overview of the main topics of process-aware information systems. However, one should be aware that the field of process-aware information systems is rather large; hence, the depth and scope of a “unified” vision should be seen in perspective. This book provides an excellent start for students and researchers interested in this topic and wishing to pursue further research. What should be remarked as a positive feature as well is the fact that each chapter provides a set of exercises at the end of each chapter. The content of the book is timely and written by internationally well-known experts in the field. The chapters are edited carefully and are well-written and easy to follow. One drawback of this book, however, is its price (some \$99), which might be challenging as a student textbook.

Schahram Dustdar
Vienna University of Technology, Austria,
Distributed Systems Group (DSG),
Institute of Information Systems,
A-1040 Wien, Argentinierstrasse 8/184-1,
Austria
E-mail address: dustdar@infosys.tuwien.ac.at
URL: <http://www.infosys.tuwien.ac.at/Staff/sd>

Available online 28 February 2006

doi:10.1016/j.ipm.2006.01.005

C. Nass, S. Brave, *Wired for Speech: How Voice Activates and Advances the Human–Computer Relationship*, The MIT Press, Cambridge, MA, 2005, 296 pp., ISBN: 0-262-14092-6, \$32.50

Humans are forever social. We humans live in social environments and behave socially. We treat everything in our environment, including other humans and even artifacts, socially. Consequently, computers and other technologies should be continuously regarded as social actors. This means that designers are challenged to develop more socially demanding interfaces for today’s technologies. Yet how much do we understand about ourselves as social beings, especially when we interact with technologies? In addition, how can our understandings help in designing and using socially friendly technologies?

Nass and Brave’s new book, *Wired for Speech: How Voice Activates and Advances the Human–Computer Relationship*, is the first of its kind to address the social aspect of human–technology interaction via one of the most social cues: speech. They assert that the psychology of interface speech is the psychology of human speech. The book’s deep theoretical roots and strong empirical evidence challenge our common understanding