Christoph Bussler
Ming-Chien Shan (Eds.)

LNCS 3811

Technologies for E-Services

6th International Workshop, TES 2005
Trondheim, Norway, September 2005
Revised Selected Papers

Springer

Christoph Bussler   Ming-Chien Shan (Eds.)

Technologies for E-Services

6th International Workshop, TES 2005
Trondheim, Norway, September 2-3, 2005
Revised Selected Papers

Springer

Volume Editors

Christoph Bussler
Cisco Systems, Inc.
3600 Cisco Way, MS SJC18/2/4, San Jose, CA 95134, USA
E-mail: chbussler@aol.com

Ming-Chien Shan
HP Labs
13264 Glasgow Court, Saratoga, CA 95070, USA
E-mail: MingChien.Shan@mail.com

Library of Congress Control Number: 2006920029

CR Subject Classification (1998): H.2, H.4, C.2, H.3, J.1, K.4.4, I.2.11

LNCS Sublibrary: SL 3 – Information Systems and Application, incl. Internet/Web and HCI

ISSN 0302-9743
ISBN-10 3-540-31067-3 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-31067-9 Springer Berlin Heidelberg New York

Springer is a part of Springer Science+Business Media

springer.com

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 11607380 06/3142 5 4 3 2 1 0

Lecture Notes in Computer Science 3811

Commenced Publication in 1973
Founding and Former Series Editors:
Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Lecture Notes in Computer Science

For information about Vols. 1–3773

please contact your bookseller or Springer

Vol. 3872: H. Bunke, A. L. Spitz (Eds.), Document Analysis Systems VII. XIII, 630 pages. 2006.

Vol. 3870: S. Spaccapietra, P. Atzeni, W.W. Chu, T. Catarci, K.P. Sycara (Eds.), Journal on Data Semantics V. XIII, 237 pages. 2006.

Vol. 3868: K. Römer, H. Karl, F. Mattern (Eds.), Wireless Sensor Networks. XI, 342 pages. 2006.

Vol. 3863: M. Kohlhase (Ed.), Mathematical Knowledge Management. XI, 405 pages. 2006. (Sublibrary LNAI).

Vol. 3861: J. Dix, S.J. Hegner (Eds.), Foundations of Information and Knowledge Systems. X, 331 pages. 2006.

Vol. 3860: D. Pointcheval (Ed.), Topics in Cryptology – CT-RSA 2006. XI, 365 pages. 2006.

Vol. 3858: A. Valdes, D. Zamboni (Eds.), Recent Advances in Intrusion Detection. X, 351 pages. 2006.

Vol. 3857: M. Fossorier, H. Imai, S. Lin, A. Poli (Eds.), Applied Algebra, Algebraic Algorithms and Error-Correcting Codes. XI, 350 pages. 2006.

Vol. 3855: E. A. Emerson, K.S. Namjoshi (Eds.), Verification, Model Checking, and Abstract Interpretation. XI, 443 pages. 2005.

Vol. 3853: A.J. Ijspeert, T. Masuzawa, S. Kusumoto (Eds.), Biologically Inspired Approaches to Advanced Information Technology. XIV, 388 pages. 2006.

Vol. 3852: P.J. Narayanan, S.K. Nayar, H.-Y. Shum (Eds.), Computer Vision - ACCV 2006, Part II. XXXI, 977 pages. 2005.

Vol. 3851: P.J. Narayanan, S.K. Nayar, H.-Y. Shum (Eds.), Computer Vision - ACCV 2006, Part I. XXXI, 973 pages. 2006.

Vol. 3850: R. Freund, G. Păun, G. Rozenberg, A. Salomaa (Eds.), Membrane Computing. IX, 371 pages. 2006.

Vol. 3848: J.-F. Boulicaut, L. De Raedt, H. Mannila (Eds.), Constraint-Based Mining and Inductive Databases. X, 401 pages. 2006. (Sublibrary LNAI).

Vol. 3847: K.P. Jantke, A. Lunzer, N. Spyratos, Y. Tanaka (Eds.), Federation over the Web. X, 215 pages. 2006. (Sublibrary LNAI).

Vol. 3844: J.-M. Bruel (Ed.), Satellite Events at the MoDELS 2005 Conference. XIII, 360 pages. 2006.

Vol. 3843: P. Healy, N.S. Nikolov (Eds.), Graph Drawing. XVII, 536 pages. 2006.

Vol. 3842: H.T. Shen, J. Li, M. Li, J. Ni, W. Wang (Eds.), Advanced Web and Network Technologies, and Applications. XXVII, 1057 pages. 2006.

Vol. 3841: X. Zhou, J. Li, H.T. Shen, M. Kitsuregawa, Y. Zhang (Eds.), Frontiers of WWW Research and Development - APWeb 2006. XXIV, 1223 pages. 2006.

Vol. 3840: M. Li, B. Boehm, L.J. Osterweil (Eds.), Unifying the Software Process Spectrum. XVI, 522 pages. 2006.

Vol. 3839: J.-C. Filliâtre, C. Paulin-Mohring, B. Werner (Eds.), Types for Proofs and Programs. VIII, 275 pages. 2006.

Vol. 3838: A. Middeldorp, V. van Oostrom, F. van Raamsdonk, R. de Vrijer (Eds.), Processes, Terms and Cycles: Steps on the Road to Infinity. XVIII, 639 pages. 2005.

Vol. 3837: K. Cho, P. Jacquet (Eds.), Technologies for Advanced Heterogeneous Networks. IX, 307 pages. 2005.

Vol. 3836: J.-M. Pierson (Ed.), Data Management in Grids. X, 143 pages. 2006.

Vol. 3835: G. Sutcliffe, A. Voronkov (Eds.), Logic for Programming, Artificial Intelligence, and Reasoning. XIV, 744 pages. 2005. (Sublibrary LNAI).

Vol. 3834: D.G. Feitelson, E. Frachtenberg, L. Rudolph, U. Schwiegelshohn (Eds.), Job Scheduling Strategies for Parallel Processing. VIII, 283 pages. 2005.

Vol. 3833: K.-J. Li, C. Vangenot (Eds.), Web and Wireless Geographical Information Systems. XI, 309 pages. 2005.

Vol. 3832: D. Zhang, A.K. Jain (Eds.), Advances in Biometrics. XX, 796 pages. 2005.

Vol. 3831: J. Wiedermann, G. Tel, J. Pokorný, M. Bieliková, J. Štuller (Eds.), SOFSEM 2006: Theory and Practice of Computer Science. XV, 576 pages. 2006.

Vol. 3829: P. Pettersson, W. Yi (Eds.), Formal Modeling and Analysis of Timed Systems. IX, 305 pages. 2005.

Vol. 3828: X. Deng, Y. Ye (Eds.), Internet and Network Economics. XVII, 1106 pages. 2005.

Vol. 3827: X. Deng, D.-Z. Du (Eds.), Algorithms and Computation. XX, 1190 pages. 2005.

Vol. 3826: B. Benatallah, F. Casati, P. Traverso (Eds.), Service-Oriented Computing - ICSOC 2005. XVIII, 597 pages. 2005.

Vol. 3824: L.T. Yang, M. Amamiya, Z. Liu, M. Guo, F.J. Rammig (Eds.), Embedded and Ubiquitous Computing – EUC 2005. XXIII, 1204 pages. 2005.

Vol. 3823: T. Enokido, L. Yan, B. Xiao, D. Kim, Y. Dai, L.T. Yang (Eds.), Embedded and Ubiquitous Computing – EUC 2005 Workshops. XXXII, 1317 pages. 2005.

Vol. 3822: D. Feng, D. Lin, M. Yung (Eds.), Information Security and Cryptology. XII, 420 pages. 2005.

Vol. 3821: R. Ramanujam, S. Sen (Eds.), FSTTCS 2005: Foundations of Software Technology and Theoretical Computer Science. XIV, 566 pages. 2005.

Vol. 3820: L.T. Yang, X.-s. Zhou, W. Zhao, Z. Wu, Y. Zhu, M. Lin (Eds.), Embedded Software and Systems. XXVIII, 779 pages. 2005.

Vol. 3819: P. Van Hentenryck (Ed.), Practical Aspects of Declarative Languages. X, 231 pages. 2005.

Vol. 3818: S. Grumbach, L. Sui, V. Vianu (Eds.), Advances in Computer Science – ASIAN 2005. XIII, 294 pages. 2005.

Vol. 3817: M. Faundez-Zanuy, L. Janer, A. Esposito, A. Satue-Villar, J. Roure, V. Espinosa-Duro (Eds.), Nonlinear Analyses and Algorithms for Speech Processing. XII, 380 pages. 2006. (Sublibrary LNAI).

Vol. 3816: G. Chakraborty (Ed.), Distributed Computing and Internet Technology. XXI, 606 pages. 2005.

Vol. 3815: E.A. Fox, E.J. Neuhold, P. Premsmit, V. Wuwongse (Eds.), Digital Libraries: Implementing Strategies and Sharing Experiences. XVII, 529 pages. 2005.

Vol. 3814: M. Maybury, O. Stock, W. Wahlster (Eds.), Intelligent Technologies for Interactive Entertainment. XV, 342 pages. 2005. (Sublibrary LNAI).

Vol. 3813: R. Molva, G. Tsudik, D. Westhoff (Eds.), Security and Privacy in Ad-hoc and Sensor Networks. VIII, 219 pages. 2005.

Vol. 3811: C. Bussler, M.-C. Shan (Eds.), Technologies for E-Services. VIII, 127 pages. 2006.

Vol. 3810: Y.G. Desmedt, H. Wang, Y. Mu, Y. Li (Eds.), Cryptology and Network Security. XI, 349 pages. 2005.

Vol. 3809: S. Zhang, R. Jarvis (Eds.), AI 2005: Advances in Artificial Intelligence. XXVII, 1344 pages. 2005. (Sublibrary LNAI).

Vol. 3808: C. Bento, A. Cardoso, G. Dias (Eds.), Progress in Artificial Intelligence. XVIII, 704 pages. 2005. (Sublibrary LNAI).

Vol. 3807: M. Dean, Y. Guo, W. Jun, R. Kaschek, S. Krishnaswamy, Z. Pan, Q.Z. Sheng (Eds.), Web Information Systems Engineering – WISE 2005 Workshops. XV, 275 pages. 2005.

Vol. 3806: A.H. H. Ngu, M. Kitsuregawa, E.J. Neuhold, J.-Y. Chung, Q.Z. Sheng (Eds.), Web Information Systems Engineering – WISE 2005. XXI, 771 pages. 2005.

Vol. 3805: G. Subsol (Ed.), Virtual Storytelling. XII, 289 pages. 2005.

Vol. 3804: G. Bebis, R. Boyle, D. Koracin, B. Parvin (Eds.), Advances in Visual Computing. XX, 755 pages. 2005.

Vol. 3803: S. Jajodia, C. Mazumdar (Eds.), Information Systems Security. XI, 342 pages. 2005.

Vol. 3802: Y. Hao, J. Liu, Y.-P. Wang, Y.-m. Cheung, H. Yin, L. Jiao, J. Ma, Y.-C. Jiao (Eds.), Computational Intelligence and Security, Part II. XLII, 1166 pages. 2005. (Sublibrary LNAI).

Vol. 3801: Y. Hao, J. Liu, Y.-P. Wang, Y.-m. Cheung, H. Yin, L. Jiao, J. Ma, Y.-C. Jiao (Eds.), Computational Intelligence and Security, Part I. XLI, 1122 pages. 2005. (Sublibrary LNAI).

Vol. 3799: M. A. Rodríguez, I.F. Cruz, S. Levashkin, M.J. Egenhofer (Eds.), GeoSpatial Semantics. X, 259 pages. 2005.

Vol. 3798: A. Dearle, S. Eisenbach (Eds.), Component Deployment. X, 197 pages. 2005.

Vol. 3797: S. Maitra, C. E. V. Madhavan, R. Venkatesan (Eds.), Progress in Cryptology - INDOCRYPT 2005. XIV, 417 pages. 2005.

Vol. 3796: N.P. Smart (Ed.), Cryptography and Coding. XI, 461 pages. 2005.

Vol. 3795: H. Zhuge, G.C. Fox (Eds.), Grid and Cooperative Computing - GCC 2005. XXI, 1203 pages. 2005.

Vol. 3794: X. Jia, J. Wu, Y. He (Eds.), Mobile Ad-hoc and Sensor Networks. XX, 1136 pages. 2005.

Vol. 3793: T. Conte, N. Navarro, W.-m.W. Hwu, M. Valero, T. Ungerer (Eds.), High Performance Embedded Architectures and Compilers. XIII, 317 pages. 2005.

Vol. 3792: I. Richardson, P. Abrahamsson, R. Messnarz (Eds.), Software Process Improvement. VIII, 215 pages. 2005.

Vol. 3791: A. Adi, S. Stoutenburg, S. Tabet (Eds.), Rules and Rule Markup Languages for the Semantic Web. X, 225 pages. 2005.

Vol. 3790: G. Alonso (Ed.), Middleware 2005. XIII, 443 pages. 2005.

Vol. 3789: A. Gelbukh, Á. de Albornoz, H. Terashima-Marín (Eds.), MICAI 2005: Advances in Artificial Intelligence. XXVI, 1198 pages. 2005. (Sublibrary LNAI).

Vol. 3788: B. Roy (Ed.), Advances in Cryptology - ASIACRYPT 2005. XIV, 703 pages. 2005.

Vol. 3787: D. Kratsch (Ed.), Graph-Theoretic Concepts in Computer Science. XIV, 470 pages. 2005.

Vol. 3785: K.-K. Lau, R. Banach (Eds.), Formal Methods and Software Engineering. XIV, 496 pages. 2005.

Vol. 3784: J. Tao, T. Tan, R.W. Picard (Eds.), Affective Computing and Intelligent Interaction. XIX, 1008 pages. 2005.

Vol. 3783: S. Qing, W. Mao, J. Lopez, G. Wang (Eds.), Information and Communications Security. XIV, 492 pages. 2005.

Vol. 3782: K.-D. Althoff, A. Dengel, R. Bergmann, M. Nick, T.R. Roth-Berghofer (Eds.), Professional Knowledge Management. XXIII, 739 pages. 2005. (Sublibrary LNAI).

Vol. 3781: S.Z. Li, Z. Sun, T. Tan, S. Pankanti, G. Chollet, D. Zhang (Eds.), Advances in Biometric Person Authentication. XI, 250 pages. 2005.

Vol. 3780: K. Yi (Ed.), Programming Languages and Systems. XI, 435 pages. 2005.

Vol. 3779: H. Jin, D. Reed, W. Jiang (Eds.), Network and Parallel Computing. XV, 513 pages. 2005.

Vol. 3778: C. Atkinson, C. Bunse, H.-G. Gross, C. Peper (Eds.), Component-Based Software Development for Embedded Systems. VIII, 345 pages. 2005.

Vol. 3777: O.B. Lupanov, O.M. Kasim-Zade, A.V. Chaskin, K. Steinhöfel (Eds.), Stochastic Algorithms: Foundations and Applications. VIII, 239 pages. 2005.

Vol. 3776: S.K. Pal, S. Bandyopadhyay, S. Biswas (Eds.), Pattern Recognition and Machine Intelligence. XXIV, 808 pages. 2005.

Vol. 3775: J. Schönwälder, J. Serrat (Eds.), Ambient Networks. XIII, 281 pages. 2005.

Vol. 3774: G. Bierman, C. Koch (Eds.), Database Programming Languages. X, 295 pages. 2005.

Preface

The 6th Workshop on Technologies for E-Services (TES-05) was held September 2-3, 2005, in conjunction with the 31st International Conference on Very Large Data Bases (VLDB 2005) in Trondheim, Norway.

The next generation of applications will be developed in the form of services that are offered over a network, either a company's intranet or the Internet. Service-based architectures depend on an infrastructure that allows service providers to describe and advertise their services and service consumers to discover and select the services that best fulfill their requirements. Frameworks and messaging protocols for e-services in stationary and mobile environments are being developed and standardized, metadata and ontologies are being defined, and mechanisms are under development for service composition, delivery, monitoring, and payment. End-to-end security and quality of service guarantees will be essential for the acceptance of e-services. As e-services become pervasive, e-service management will play a central role.

The workshop's objective is to provide a forum for researchers and practitioners to present new developments and experience reports. The goal of the TES workshop is to identify the technical issues, models, and infrastructures that enable enterprises to provide e-services to other businesses and individual customers.

In response to the call for submissions, 40 papers were submitted, out of which the Program Committee selected 10 high-quality submissions for presentation at the workshop. Unfortunately, one author had to withdraw, and the remaining nine papers that were presented are included in these proceedings.

The workshop itself started with a keynote given by Umeshwar Dayal on "Optimization of Business Processes (& Composite E-Services)." He gave interesting insights into past, current, and future developments in the area of business process management and the application of these results to composite e-services. Afterwards the three technical sessions "Design," "Technology," and "Composite Web Services" took place and very interesting and stimulating research paper presentations were delivered, resulting in many questions and discussions.

We would like to thank the authors for their submissions to the workshop, the Program Committee for their hard work during a very brief reviewing period, the keynote speaker and presenters for their very interesting presentations, and the audience for their interest, questions, and discussions.

September 2004

Christoph Bussler
Ming-Chien Shan

Workshop Organization Committee

Christoph Bussler
Digital Enterprise Research Institute (DERI), Ireland

Ming-Chien Shan
Hewlett-Packard Laboratories, CA, USA

Program Committee

Boualem Benatallah
Fabio Casati
Emilia Cimpian
Jen-Yao Chung
Francisco Curbera
Marlon Dumas
Schahram Dustdar
Timothy Finin
Dimitrios Georgakopoulos
Paul Grefen
Juan Gomez
Armin Haller
Manfred Hauswirth
Kashif Iqbal
Mick Kerrigan
Jacek Kopecky
Frank Leymann
Adrian Mocan
Heiko Ludwig
Eyal Oren
Cesare Pautasso
Mike Papazoglou
Barbara Pernici
Krithi Ramamritham
Manfred Reichert
Dumitru Roman
Brahmananda Sapkota
Karsten Schulz
Amit Sheth
Thomas Strang
Katia Sycara
Farouk Toumani
Kunal Verma
Michal Zaremba
Liang-Jie Zhang
Michal Zaremba
Lizhu Zhou

Table of Contents

Challenges in Business Process Analysis and Optimization

Malu Castellanos, Fabio Casati, Mehmet Sayal, and Umeshwar Dayal

Hewlett-Packard Laboratories, Palo Alto, CA
firstname.lastname@hp.com

Abstract. As organizations become more and more process-centered and as the extended enterprise becomes more of a reality, there is an increasing need to move towards the intelligent enterprise characterized by its agility for provisioning resources and services and the adaptation and optimization of its business processes. Many are the challenges that need to be faced to equip the extended enterprise with intelligence, in particular with business process intelligence. In this paper we present an overview of some challenges, opportunities and directions for analysis and optimization of business processes towards the intelligent enterprise.

1 Business Process Intelligence

Over the years the concept of *enterprise* has evolved dramatically as its functionalities and boundaries have extended in response to business and technological changes. The traditional monolithic enterprise was characterized by all of its operations being performed in-house, and as a consequence, the key challenges addressed by the enterprise in the 1980's were functional automation and enterprise integration. Competitive pressures in the 1990's and early 2000's have compelled enterprises to outsource many of their non-core operations such as financial processing or even manufacturing. Advances in network technology, information exchange standards and middleware have made it possible to reach beyond the boundaries of the traditional enterprise to integrate trading partners, suppliers, and service providers giving rise to the *extended* enterprise. B2B frameworks and protocols (such as Oasis/ebXML and RosettaNet) have emerged for partner integration and the management of collaborative business processes across the extended enterprise. The high complexity of the extended enterprise has made it crucial to devise ways of operating and managing its components and interactions in an automatic, efficient and flexible way. In particular, integration of information and process flows, agile provisioning of resources and services, monitoring, analysis, adaptation and optimization of business processes are characteristic of a new concept of enterprise, the *intelligent* enterprise.

There are many challenges to make the enterprise transformation a reality. Architecture and infrastructure for its ecosystem; modeling of collaborative, inter-organizational business processes that span the partners in the extended enterprise; description, composition, publishing, registration and discovery of business service providers; rapid and dynamic connection and disconnection of partners; security and trust; and negotiation to

C. Bussler and M.-C. Shan (Eds.): TES 2005, LNCS 3811, pp. 1 – 10, 2006.

name a few. In this paper, we focus on aspects related to the management and optimization of business processes in the intelligent enterprise.

Our key thesis in this paper is that an (extended) enterprise is an entity that delivers services to customers, and it is the business processes underlying such services which drive the enterprise (Figure 1). Traditional business process management supports the definition, execution and monitoring of business processes, defines and controls the sequence, timing and other dependencies among tasks, enforces business rules and policies, assigns resources (humans, data, applications, web services) to individual tasks, monitors and tracks all aspects of the process execution. Inter-enterprise business process management extends the above functionalities to the extended enterprise as a result of the marriage between process management and web services technologies. In the intelligent enterprise business process management is equipped with functionalities that facilitate the optimization of the quality (in terms of metrics meaningful to the enterprise –internal quality-, or to the customers –external quality-) of intra- and inter-enterprise business processes. This is known as *Business Process Intelligence.*

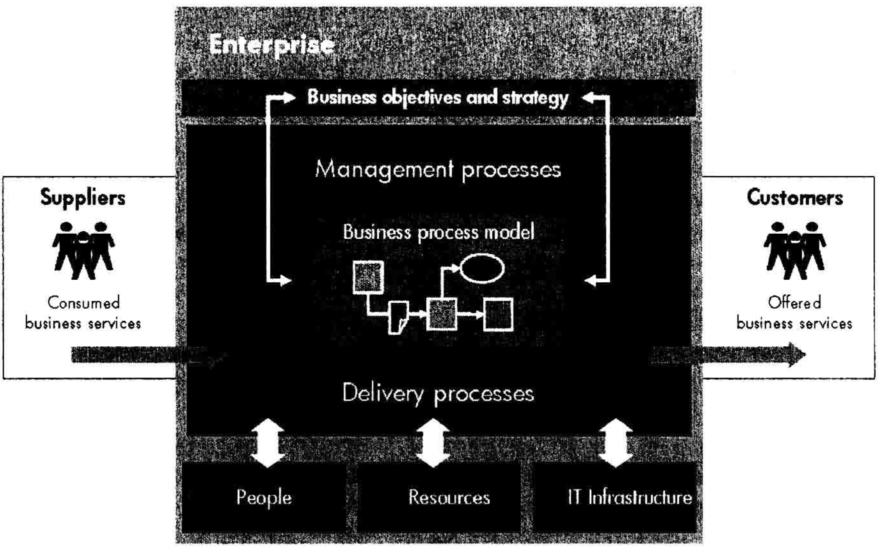

Fig. 1. Business processes drive the Enterprise

In business process intelligence, quality metrics can be computed from process execution data logged in various forms (workflow engine logs, web service execution logs, application logs, event logs, etc). For example, the performance metric of an order process could be derived from the start and end times of execution of orders processed during a given period of time. By warehousing [Bonifati01] the execution data and the metrics the business process can be monitored, analyzed and optimized with different kinds of techniques relying on data mining, statistics, simulation, and optimization, among others [Castellanos05C]. The purpose of this paper is to discuss opportunities for analysis and optimization of business processes along with some directions and challenges. In particular, we discuss six challenging areas: process validation, resource allocation, service selection, work queue management, time-based correlation, and prediction and proactive optimization.

2 Process Validation

Business processes underlying enterprise business operations often start with a model (composed by individual tasks along with their order and dependencies) that is mapped to the IT infrastructure to be implemented accordingly. Unfortunately, it is often the case that the initial alignment of the process model and its implementation doesn't last too long, sooner or later the implementation starts to deviate from the model in response to changes in the infrastructure or in the business objectives of the enterprise. New steps and even paths can be added to handle new situations and exceptions, conditions in the decisions may be modified to adapt to new requirements for following one path or another, steps could be merged in response to some changes in the organization, and so on. At some point nobody knows what is the 'real' process and what deviations have taken place. Moreover, the deviations can occur only for some process executions without anybody knowing in which cases or contexts this occurs.

The misalignment between the real process and its original model needs to be resolved in a number of situations. To optimize a process or simply improve its operation, it is necessary to have a clear picture of the model as it is executed in practice. When the process has been subject to changes over time, it is even more important to analyze how such changes or deviations are affecting its performance and what are the opportunities to improve it. In addition, new legislations such as Sarbanes-Oxley, require to document processes and add controls where appropriate. For this, it is necessary to find the deviations that have happened and obtain a process model that corresponds to the reality. Monitoring is yet another activity that requires checking the consistency between the analyst's high level view and the actual process. Finally, by identifying process model deviations it is possible to perform sensitivity analysis to analyze alternatives to such deviations.

Triggered by situations like the aforementioned, research has started in this direction under the name of process validation, process alignment, delta analysis and conformance testing. It basically consists of assessing whether a process model actually corresponds to what is executed by a company's IT infrastructure and in identifying the deviations if that is not the case. This is related to the area of process discovery which elicits the process model from the logged execution data when such a model does not exist. In contrast, in process validation the model already exists and it is the deviations from it what is of interest. In both cases it is the event, transaction or audit log what is mined to discover how the process is being conducted in reality. This kind of log typically records events that correspond to the start and completion of activities in the IT infrastructure. In addition, the log entry provides information about the task identifier and about the process execution identifier in the context of which the task is being performed. As an example, an entry in the log could say that an 'inventory check' step started at 3.30pm and completed at 4.15 pm, and that it has been executed in the context of order processing #539 (where 'order processing' is the process type whose model we want to verify, while #539 identifies a specific execution - also called case or instance - of the process). Different products like WFMS, ERPs, transaction monitor systems like HP OVTA, HP BPI, to name a few, provide this kind of log.

A few techniques have been proposed to validate processes and most of them consist of applying process discovery to learn the 'real' model [Aalst03] and then compare it with the original one to identify deviations and measure the degree of (mis)alignment [Rozinat05]. The challenge remains in developing alternative techniques that do not require discovering the real model so that the cost of verification is lower while at the same time gaining more efficiency. Furthermore, to make things more versatile and automatic, research needs to be done for techniques that could keep track of deviations from a process model as they occur, associate them to their context, and suggest changes to the model as more evidence of these deviations is collected.

3 Resource Allocation

A business process is formally defined as a sequence of tasks which are carried out by resources. The resources that are in charge of executing the individual process tasks can be humans, software applications, or web services. The allocation of resources to tasks can be done in either static or dynamic manner. Static resource allocation means that a pre-defined set of assignments between resources and tasks is applied and stays fixed for all instances (executions) of the business process. A dynamic allocation allows the process designers to define resource pools, which contain resources that are capable of carrying out a particular task or a set of tasks, and the actual assignment is done during the execution of each individual process instance. The concept of *Virtual Resources* is generally used for defining such resource pools.

The allocation of resources to tasks can significantly affect the performance and outcome of the business processes, which in turn affects the quality of services and products of an enterprise. It is possible to allocate one or more resources from a virtual resource pool to a particular task. Identification of bottlenecks in a business process and proper allocation of resources to critical tasks can help a business meet the deadlines and Service Level Agreement (SLA) terms while delivering services and products at a desired quality.

Business process simulation tools are used for analyzing the behavior of resources and their affect on the overall performance and outcome of processes. Businesses often need to know what would be the effect on business goals (in terms of metrics) if certain process parameters would be set to different values. For example, it is important to know (or predict) the effect of assigning two resources to a particular task, instead of one. Sensitivity analysis (what-if analysis) allows users to analyze outcomes of various scenarios in which the effect of different parameter settings can be observed. It is important to know how much benefit such an additional resource allocation could bring. Possible parameters for simulation and sensitivity analysis could be resource pool sizes for individual tasks, inter-arrival rate of entities to be processed, resource behavior (response time to particular tasks), and cost of individual resources (per unit time or total). It is usually assumed that data needed for setting up simulation parameters is available from audit logs of Workflow Management Systems (WFMS). Most business process definitions in WFMS tools include explicit information about resource-task allocation. However, business processes are not always automated using

WFMS, and such allocation information is not usually available. Therefore, it is often necessary to extract resource allocation manually or automatically using various data analysis and mining techniques.

Businesses are interested not only in understanding the effect of changes in a process but also in determining the best possible allocation of resources in order to achieve certain performance and quality goals. Simulation leveraged with a search technique offers one solution [Castellanos05C]. Here, the objective could be to minimize the number of process instances that exceed a certain metric threshold or to maximize the overall value of a given metric. Other metrics that need to be kept within range would be the constraints and the maximum number of units in each resource pool would constitute the bounds of the search space. The challenge is how to take advantage of domain knowledge to quickly reduce the search space to minimize the number of resource allocation configurations that need to be simulated. Other techniques, such as mathematical programming, can be brought to bear on this problem.

It is also important to automatically identify the resources that perform poorly in certain contexts. Data mining, and in particular classification algorithms, can be used for this purpose [Casati04]. For example, a classification rule may indicate that involvement of a particular resource in task executions results in poor performance. Replacement of such a resource with another one from a resource pool may yield much better execution performance. Similarly, time-correlation detection [Sayal05] can tell us the time-dependent cause-effect relationships among any numeric variables (e.g., business metrics or operational data). As an example, a decrease in the performance of a particular resource may trigger the appearance of a bottleneck in a given critical step of a process after a certain time period. Automatic detection of such time-correlations can help business managers optimize resource allocation to proactively avoid such effects.

4 Service Selection

A related issue that can potentially affect the quality or performance of a business process is that of dynamic service selection, that is, of selecting at run-time the service that can best perform (implement) a certain step in the process. The lack of maturity and of standardization at the business level (that is, lack of standard business interfaces, protocols, and in general of high-level B2B standards) has made this problem to date irrelevant for the industry, where processes had to be implemented with a static binding to the services implementing the steps. However, lately we are witnessing an increased push not only to standardize low-level interaction protocols, but also protocols (and semantics) at the business level (e.g., what is the meaning of a *PurchaseOrder* document and how it should be exchanged among parties to arrive at a complete purchase transaction). This raises the possibility, and hence the opportunity, for dynamic service binding, at least between closed communities of trusted partners. Ideally, the goal is to select, at each step, the service that maximizes the probability of achieving certain quality of performance goals.

Many approaches to this problem exist, but they are mostly based on having providers define non-functional parameters that specify the cost, duration, and other *service quality metrics*. Then, for each step in the process execution, a choice is made about the provider to be used by matching desired and provided quality, or by computing some utility function over the service quality metrics, with the goal of achieving the best possible process execution "quality". This kind of approaches has several limitations. For example, it requires clients (the ones executing the process) to trust the accuracy of the metric values declared by the providers, it assumes the ability to identify the important service metrics that contribute to high process execution quality, it requires a standard way to model metrics so that customer and providers can understand each other, and it assumes that customers have a clear idea of how exactly process quality goals depend on (are a function of) the value of the providers' promised quality parameters. In general, these are not realistic assumptions. Hence, the challenge here is how to perform dynamic service selection without incurring in the limiting assumptions described above (which also require significant modeling efforts).

A possible solution consists in having customers define the quality goals of their business processes and then specify which past executions they consider to be of high quality. This can be done by either specifying a function over process execution data that labels process executions with quality measures (e.g., all invoice payment processes that completed before the payment due date are "good"), or by having customers explicitly label process executions, for example based on feedback collected online. Once the quality of past process executions are known or measurable, quality-labeled process execution data are mined to build a set of models that, at each stage during process execution and based on the specific characteristics of each process execution, identify the service provider that historically, in ''analogous'' situations, has contributed to high quality processes. This approach is feasible since it focuses on what users really care about, what is the overall quality of the process; it involves little modeling effort (defining process quality goals is the only effort needed by the customer); and it is not based on provider-defined non-functional parameters for each service, but rather based on facts, and can progressively adjust based on changing notions of process quality as well as on changing behavior of service providers. We refer the interested reader to [Casati04] for details.

5 Work Queue Management

Many business processes are characterized by manual steps, that is, steps executed by human users (called processors) rather than services. In some cases the step creates a work item which is assigned to a specific processor, while in others the item is placed in a shared work queue.

In both cases, processors have, at any given time, several work items on which they can work, often belonging to different process instances, and they typically have flexibility in selecting which one to process first. The problem here is how to find out which is the "optimal" order in which work items should be executed, where by optimal we mean the order that minimizes SLA violations, or the total penalties resulting from the SLA violations, or that optimize some user-defined performance function (that can for example take into account SLA violations, penalties, importance of the customers, etc..). Ideally, we are looking for techniques that prioritize the items in the work queues to optimize this user-defined function. For simplicity, we just assume that the goal is that of minimizing deadline expirations for the processes.

Leaving the decision of which work item to pick completely to the processor may lead to adverse consequences and to violations of SLAs. For example, the processors may start working on an item belonging to a process instance that is, performance-wise, on target, and neglect more urgent work.

Identifying such an optimal solution is however not trivial, and indeed simple approaches do not solve the problem. For example, simply selecting the "oldest" work item would be an incorrect approach, as SLA violations depend on the process duration not on the duration of each step. Even if the work item has been started long ago, we may be well in time to meet our deadlines. Selecting the work item corresponding to the instance closer to reaching its deadline also does not produce the desired result: if we are very close to the deadline already, we won't be able to finish in time anyway, and we may as well give higher priorities to work items of other process instances. In general, we have therefore to be aware of the expected durations of the forthcoming steps to assess what priority to give to a work item, and assess which priority order minimizes violations. Depending on the level of sophistication, we may use a simple average to derive the expected time, or we may use complex forecasting techniques that based on the characteristics of the specific process instance at hand, try to predict the duration of forthcoming steps (e.g., approval for large orders typically take longer than those for small orders).

Finally, we observe that this problem is more complicated than traditional processor scheduling or job shop scheduling problems, because the process structure also comes into play. Assume that our process has two parallel branches that synchronize at some point in the flow. There is no point in having one of the two advance fast if the other is progressing slowly. Hence, to order work items, we need to be aware of what is happening in the different branches of a process instance and make sure that all branches progress (or all branches do not progress, and we leave higher priority to other instances).

6 Time-Based Correlation

Another key problem in business process intelligence is that of relating metrics on the process to operational data such as the performance of IT resources that participate in the implementation of the process. This is important because it enables understanding of the *business impact* of operational problems.

Time-correlation algorithms can be used for automatic detection of time-dependent relationships among business metrics and operational data (e.g., the execution time of tasks in a process or performance of individual resources). In [Sayal05] the input to the algorithm is a set of numeric data streams, each of which contains the recorded numeric values of one variable (business metric or operational data variable) over the course of time. The output of the algorithm is a set of time-correlation rules that explain time-dependent relationships among data streams (numeric variables).

Detection of time-dependent relationships among hundreds or thousands of numeric variables is a difficult task that needs to be achieved in order to understand the cause-effect relationships among various events in a business. The method proposed in [Sayal05] transforms numeric data streams into sequences of discrete events that correspond to change points (or landmarks) in numeric values. This transformation