Loosely-Coupled Process Automation in Medical Environments

Jurate Vysniauskaite

Departement of Informatics, University of Zurich, Binzmuhlestr. 14, 8050 Zurich, Switzerland
jurate@ifi.unizh.ch

Abstract. We discuss a case study for the hospital scenario where workflow model components are distributed across various computers or devices (e.g. mobile phones, PDAs, sensors, RFID tags). By using the concept of loosely-coupled processes we want to enable computerization of manually performed ad hoc medical tasks. That could decrease time, cost and resource consumption in a hospital. The challenges of loosely coupled process composition include: the requirement to manage flexibly the process logic, the need to overcome a heterogeneous software and hardware environment, and the management of the application domain (e.g. structure of organization, resources). The main question we address is what are the functional limitations and possibilities of current process composition approaches? In response to this question we describe and discuss the capabilities of Web service technologies. In addition, we sketch a plan to overcome these limitations. Finally, we propose a process composition approach based on BPEL4WS engine that satisfies medical scenario requirements.

1 Motivation

Healthcare is one of the most complex and dynamic environments [3]. Its complexity is related to the heterogeneity of medical processes, resource management, frequently changing patient conditions, time constraints, variety of medical records, and hospital organization structure. Traditional process-oriented information systems support only homogeneous application-to-application or application-to-consumer communication that operate over relatively simple process execution schemes. These systems mainly support production workflow coordination with limited process composition and manipulation at run time [2, 14]. Interactions between heterogeneous distributed applications and processes or sub-processes, via heterogeneous interfaces, can not be freely and quickly established. In addition, any process-oriented system cannot support various (e.g. relatively short) heterogeneous dynamic processes with distributed process logic components that contain real-time constraints and frequent human interactions [1, 8]. In Figure 1 we present an application scenario that illustrates the composition and coordination complexity of processes that often occur in a medical environment. The complexities relate to the workflow composition semantics, sub-process mapping, heterogeneous nature of tasks, composition verification, and actors in a distributed environment. Each process (e.g. examination) may consist of various heterogeneous distributed tasks. That makes it difficult to provide a fully automated system. In addition, issues like privacy or security should be considered.

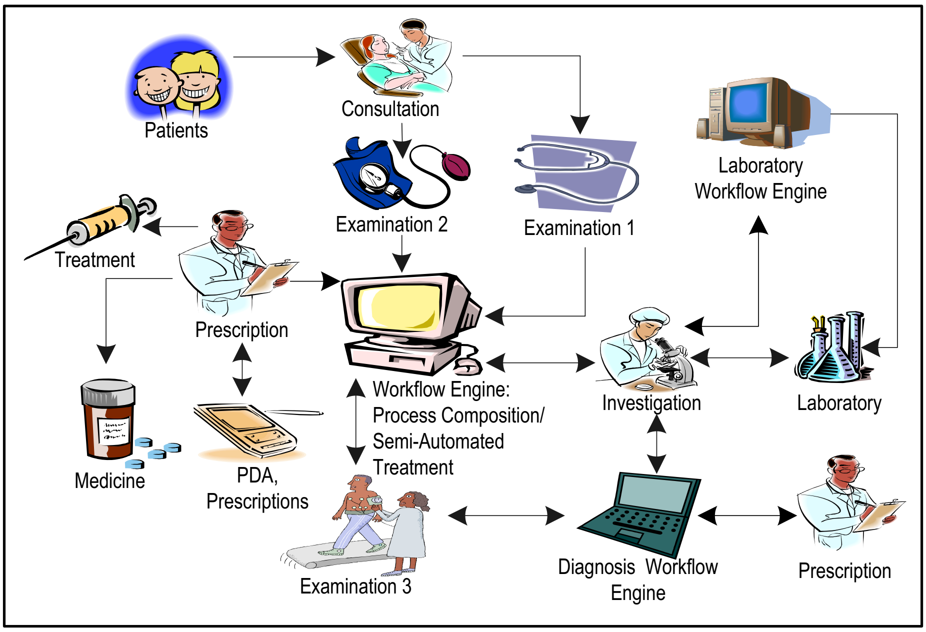

Fig. 1. *Distributed heterogeneous medical processes in a hospital scenario. Typically, each patient attends a consultation where she/he is examined, followed by a prescription and a course of treatment. Most of these processes are performed manually without support of workflow management systems. This is due to the heterogeneous nature of distributed medical processes such as examination, consultation, and treatment. This workflow composition model can process information and send results to the doctor who may examine the patient again. A workflow engine may be used to provide unified hospital services, control information flow, heterogeneous application-to-application or application-to-human communication, optimize hospital resources and arrange examinations.*

Process composition models based on message exchange (BPEL4WS, ENSEMBLE, XLANG, Oracle BPEL, SAP) [9, 18, 13] require additional specifications to enable dynamic process interactions. Those standards will be described in Section 2. Each process potentially consists of subparts that could operate as autonomous units and perform distributed tasks. For instance, a diagnosis in Fig.1 consists of a patient examination, patient health test, and diagnosis generation. These sub processes could be reused by other applications (e.g. consultation or examination). The full description of a process must define not only the behavior of each participant, but the matching of sub process patterns, to produce the overall process coordination.

The main goal of our work is to provide business process composition and coordination at run-time without the underlying coding complexities involved in integrating and modifying the process model structure. We postulate that the workflow engine should be able to "read" these process models and execute them in a platform independent manner. Our main motivation is to provide a highly automated system for medical institutions that would enable various services such as diagnosis, examinations and health monitoring using various devices (e.g. for temperature, heart, and blood measurements)[16]. One of the available perspectives is to look at how process logic (e.g. application) is distributed and implemented. This will allow us to identify workflow engine requirements where process components are distributed on various computers and devices (e.g. mobile devices, PDAs, sensors). Up till now most process-oriented systems, either commercial products or research prototypes, support processes that mainly run on centralized systems [5]. In our work, processes are performed as autonomous units and at the same time can invoke other heterogeneous distributed processes to form a new process unit. This generates complex process control flow coordination and composition schemes.

This paper investigates service-oriented technologies and outlines workflow engine requirements which could ensure dynamic health care process composition. We start with a discussion of process composition technologies (Web services). We identify the list of medical process features that require additional workflow engine functionalities. Section 3 summarizes the main requirements of message-based workflow engines [22, 13]. Then, we propose an approach for extending the process execution paradigm [19]. Section 4 outlines the work to be performed during this PhD and the research methodology we are planing to employ.

2 Background

In this section a number of different research projects and systems are studied and some of the relevant issues in related work are discussed. First, we will define the concept of a loosely-coupled process. Then we present research on various workflow engines (BPEL4WS, ENSEMBLE, SAP, Oracle BPEL) [19, 13, 16]. Finally, we discuss workflow engine limitations which motivate the requirement for the extensions for automation in loosely-coupled processes.

2.1 Business Processes in Medical Environments

The concept of a *business process* finds wide applicability in areas such as medicine, healthcare or rescue. Even if typically workflow management systems were mainly applied for operations in a bank or insurance company, the notion of business process covers a wide spectrum of tasks. The concept of a business process or process will be applied without regard to a specific purpose of a task. In the hospital scenario, Fig.1, a business process may consist of tasks such as patient condition monitoring, consultation, or history report generation. In this paper we consider a business process to be a collection of tasks which can be performed by software systems, people, and groups of people, or a combination of both [8].

Business processes can be classified according to repetition rates, business value or complexity of process logic. In this paper we are focusing on *ad hoc* [1] and *loosely-coupled* [15] processes which are prevalent in the medical environment. The *loosely-coupled process* concept appeared recently in the content of distributed system technologies (Web services). It does not have an agreed definition from the workflow management technology perspective. This term will be applied here to identify a collection of logically related tasks that can exchange data, but can also operate independently. For example, multiple outputs could be integrated into a single patient health monitoring process as inputs. We specify the loosely coupled process as a subclass of the business process that might encapsulate various collections of distributed tasks. A loosely coupled process differs from other business process categories by having the capability to map various independently implemented distributed tasks across distributed computers and devices.

2.2 Current Workflow Engines Based on Web Services

Standard workflow technologies date back to the 1980's and are based on the Workflow Coalition Model [1, 3, 6, 8, 17]. They were not designed for loosely coupled process integration and composition in a dynamic environment. Current web service composition languages including BPEL4WS, WSDL, WSFL, WS-CDL, BPML, XLANG and WS-CDL [10, 21, 20, 12] have been developed to address composition requirements. Some of these languages have well defined composition, while other provide strong workflow features. These languages support the imperative part of services composition, namely exception handling, compensation and stateful context, and all possess the capability to compose more complex structures and activities. The control structures of these languages are sufficient to model sequence, parallel, synchronization, exclusive choice and simple choice, while the BPEL4WS has the capability to model multiple choice and synchronization merge. BPEL4WS defines multiple service interactions and the coordination of imported or exported functionality via web services interactions. BPEL4WS has more powerful semantics on event handling than BPML. In transaction and state alignment support, WS-CDL is ahead of BPEL4WS. Security and reliability is not well handled in all these languages. Most recent approaches that address interoperability issues of distributed processes and applications support rather simple application-to-consumer and application-to-application communication. Web services interact with each other using XML messages [7, 4, 9, 19]. Communication protocols (SOAP) can be used to transmit XML messages. The interfaces of web services can be described in WSDL, which defines the ports that web services can connect-to. However, WSDL does not provide any information about web service behavior. Process management could be defined using standards such as BPEL4WS or its predecessor languages, XLANG and WSFL. However, the process composition, communication, coordination, validation, verification and coordination are limited by workflow engines (e.g. BPEL4WS) that carry out static interactions. A loosely coupled workflow engine typically should assume peer-to-peer message exchanges in synchronous and asynchronous modes, failure recovery, dynamic process composition with long-running communications involving many heterogeneous parties. The process composition models such as process-based, rule-based, transaction-based, and artificial intelligence-based [15, 18, 11, 7] are in early stages of development and can not provide process model reconfigurations at run-time in a dynamic heterogeneous environment. In [13, 16] reconfiguration involves writing the code which instructs the messaging engines about the process logic and conditions for its execution. That limits integration possibilities of new heterogeneous processes into the workflow model. In addition, it requires designers to develop new process model configurations, which is time and cost consuming.

3 PhD Proposal

We are going to propose an improved process composition model for loosely coupled process interaction. It will allow sub process mapping, synthesis, optimization, and verification across heterogeneous environments. This approach will enable extensions of BPEL4WS workflow engine functionalities. Our model will allow dynamic reconfiguration of subprocess interaction paths and optimize resources by applying optimization algorithms. We will perform process composition based on existing structural patterns supported by BPEL4WS [22] such as sequence, loops, or parallel execution is flexible enough to generate complex medical workflow models for dynamic compositions [11]. However, we are focusing on workflow engine extensions which are necessary to execute complex real-time distributed processes. In Figure 2 we propose extended translation of medical processes into a workflow model. This model will provide not only various medical services but will allow us to modify and integrate new services just within a few steps. In addition, it could allow process model reconfiguration at run-time.

3.1 Our Main Objectives

Our main target is dynamic composition of adaptive loosely-coupled medical processes with time constraints. The system should support communication and the sharing of information among medical personnel. The main goal is to enable the mapping of distributed heterogeneous processes and provide the right data to the right personnel at run-time. In addition, we are going to aim for optimal composition and resource allocation. The system must allow easy reconfiguration of sub-processes and correct execution flow if sub-processes can not be executed within the assigned time. To provide such interactions, flexible workflow engines are required. We will specify composition and communication of processes via message exchange (via HL7) [13, 16] between users. One of the main challenges is business process decomposition into sub processes that should be performed in a platform-independent manner [13] at run-time. Additionally, techniques are required to handle distributed process model data dependencies, coordinate inputs/outputs, specify exceptional conditions, synchronous and asynchronous interactions and their consequences including recovery sequences, and to support multiple nested process units.

3.2 Process Composition and Coordination Approach

The primary challenge in a hospital environment is that, unlike in traditional enterprise workflow architectures [8], there is no central authority that manages the business process composition of participating heterogeneous healthcare systems. Automated protocol engines [9, 13, 19] are able to track the state of simple process instances and help enforce protocol correctness in a message flow. Some loose coupling is enabled using external protocols that define the roles of the participants [19]. However, these protocols can not provide heterogeneous process composition and communication for distributed mobile parties. Such processes always involve a number of interactions with more than one heterogeneous participant.

Most of existing message-based engines [13, 9, 19] provide a subset of composition operations. Such operations include sequential and parallel control flow constructs, long running transactions, correlation of messages, handling of internal and external exceptions, dynamic service referral, and multi role contracts [22]. These patterns offer enough flexibility to define complex process models. Workflow engines restrict these execution schemes to rather simple static process scenarios in order to keep correct execution flow (BPEL4WS) [9]. In addition, process example diagnosis, as in Figure 1, illustrates the complexity of a heterogeneous loosely coupled process model. Problems arise when changes must be applied in a process flow at run-time. Therefore, a new process composition approach is required.

We will extend BPEL4WS specifications with additional process composition features, including real-time and priority constraints applied within transactions, rules, data sources, medical organization resources (e.g. medical equipment availability), and access control. We define workflow coordination as a process. This extension is necessary in order to enable communication between processes of a distributed and heterogeneous nature. Then we will propose a framework where extended process model is translated to an intermediate representation. That requires us to consider all execution paths of the defined sub-processes and provide optimal resource allocation. In addition, heterogeneous data formats should be processed and mapped. Then, the BPEL4WS script is generated. Due to enriched roles of personnel and a variety of medical tasks an extensions to the BPEL4WS execution model will be implemented. Additional optimization is required, which we apply on top of BPLE4WS scripts. This work will be based on algorithms that can collect information about all sub-process execution paths, description

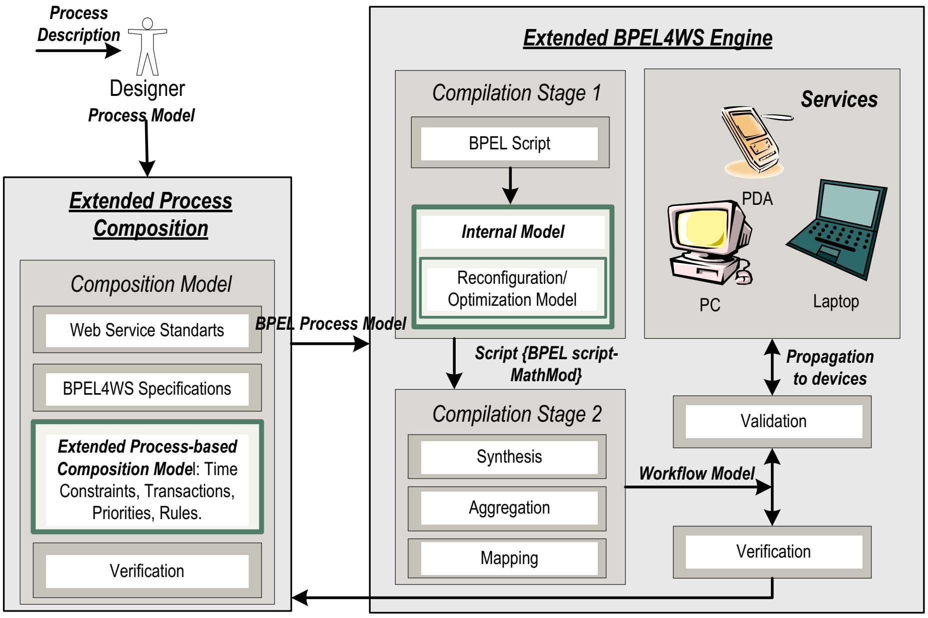

Fig. 2. *Loosely-Coupled Process Composition Approach*

of run-time modifications, priorities, time constraints, monitored events and reconfiguration history of the workflow model. Then, a second compilation stage will be performed and services provided. In case of failure, the workflow model returns to the initial process modeling stage. We believe this model will increase flexibility of message-based workflow engine functional capabilities for both static and real-time operations.

4 Discussion, further work and conclusions

The motivating objective for our work is to enable process coordination with necessary quality-of-service required in a healthcare environment. We have reviewed existing approaches for loosely coupled process composition. We have proposed a new approach for process model reconfiguration at run time. We are going to implement and test extensions of message-based workflow engines [9, 13] that could provide better distributed sub-process model configuration and composition. We believe this model will increase flexibility of message-based workflow engine operations at run-time.

In addition, we are considering the integration of processes that could be easily redesigned by medical personnel without business analyst help. Semi-automatic process matching engine will be accessed via a graphical interface. We believe our reconfiguration approach, Fig.2, applied on top of BPEL engine [9] will optimize workflow composition, verification and resource allocation.

References

1. G. Alonso and et al. WFMS: The next generation of distributed processing tools. In *Advanced Transaction Models and Architectures*. 1997.
2. P. Dadam and et al. Clinical workflows - the killer application for process-oriented information systems? In *BIS 2000*, pages 36–59, Poznan, Poland, 2000. Springer.
3. P. Dadam and et al. Towards a new dimension in clinical information processing. In *Business Proc Management*, pages 295–301, Hannover, 2000. IOS Press.
4. S. Dustdar. Caramba - A process-aware collaboration system supporting ad hoc and collaborative processes in virtual teams. *Dist and Parall Databases*, 15(1):45–66, 2004.
5. S. Dustdar and et al. Architectural concerns in distributed and mobile collaborative systems. In *SEKE*, pages 521–522, 2002.
6. R. Filho and et al. A fully distributed architecture for large scale workflow enactment. *Int. J. Cooperative Inf. Syst*, 12(4):411–440, 2003.
7. H. Foster and et al. Using a rigorous approach for engineering web service compositions: A case study. In *IEEE SCC*, pages 217–224, 2005.
8. D. Georgakopoulos and et al. An overview of workflow management: From process modeling to workflow automation infrastructure. *Dist and Parall Databases*, 3:119–153, 1995.
9. Houspanossian and et al. Extending a BPEL engine with AOP. In *(ASSE'05)*, Rosario, Argentina, August 2005.
10. F. Curbera Ibm, H. Dholakia, and et al. Business process execution language for web services, May 13 2003.
11. M. Jaeger and et al. QoS aggregation for web service composition using workflow patterns. In *EDOC*, pages 149–159, 2004.
12. N. Kavantzas, D. Burdett, and et al. Web services choreography description language version 1.0. World Wide Web Consortium, Candidate Recommandation CR-ws-cdl-10-20051109, November 2005.
13. J. Klein. Ensemble: Integrating electronic health records using HL7 clinical document architecture. *Business Integration Journal*, 2005.
14. K. Kuhn and et al. A conceptual approach to an open hospital information system. Technical report, Ulm, 1993.
15. N. Lakhal and et al. THROWS: An architecture for highly available distributed execution of web services compositions. In *RIDE*, pages 103–110, 2004.
16. M. LaRocca. The overlooked role of abstraction and data persistence in service oriented architectures. Technical report, InterSystems, May 2005.
17. S. Rinderle and et al. Evaluation of correctness criteria for dynamic workflow changes. In *Business Proc Management*, pages 41–57, 2003.
18. B. Shafiq and et al. A GTRBAC based system for dynamic workflow composition and management. In *ISORC*, pages 284–290, 2005.
19. S. Tai and et al. Composition of coordinated Web services. *Springer LNCS:3231*, pages 294–300, 2004.
20. S. Thatte. XLANG web services for business process design, 2001.
21. S. A. White et al. Business process modeling notation (BPML), working draft, version 0.9, 2002.
22. P. Wohed and et al. Pattern-based analysis of BPEL4WS. QUT Technical report, FIT-TR-2002-04, Queensland University of Technology, Brisbane, 2002.