Making Business Processes Adaptive Through Semantically Enhanced Workflow Descriptions

Herwig Zeiner Wolfgang Halb Harald Lernbeiß

Bernhard Jandl Christian Derler

JOANNEUM RESEARCH
Institute of Information Systems
Graz, Austria
firstname.lastname@joanneum.at

ABSTRACT

The research presented in this paper describes a methodology for making business processes described in BPEL adaptive by using semantically enhanced workflow descriptions. Based on widely used technologies for business process management in service oriented architectures new methods are introduced. They enable flexible process execution and service selection while maintaining the required level of resilience and security in such sensitive settings. An evaluation of the system's performance shows that the overhead introduced by the dynamic service selection is minor, given that the impact of adaptability on the entire process execution is extremely positive.

Categories and Subject Descriptors

H.3.5 [**Information Storage and Retrieval**]: Online Information Services

General Terms

Design, Performance, Reliability

Keywords

adaptive processes, semantic business process management, service-oriented architectures, semantic web services

1. SERSCIS APPROACH FOR ADAPTIVE PROCESSES

Adaptivity of business processes is an important function that enables businesses to remain productive and limit disruptions to vital processes even under changing and challenging circumstances. Especially in the area of critical infrastructures it is essential to keep business processes alive

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

I-SEMANTICS 2010, September 1-3, 2010 Graz, Austria

Copyright 2010 ACM 978-1-4503-0014-8/10/09 ...$10.00.

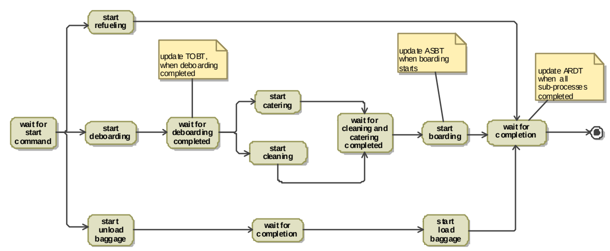

Figure 1: Turn-round process

and adapt to service disruptions and other factors that may occur at any time. The SERSCIS ("Semantically Enhanced, Resilient and Secure Critical Infrastructure Services") project[1] aims at supporting the use of interconnected information and communication technologies systems that underpin critical infrastructures. The approach presented in this paper is a part of the SERSCIS project that deals with the adaptivity of business processes by using semantically enhanced workflow descriptions that can be seen as an extremely lightweight variation of semantic web services. In contrast to conventional workflow execution systems, the workflow templates can be annotated semantically. Several smaller steps are involved in such adaptive workflow processes such as service matching, selection and replacement of services. Matching is the process of finding an appropriate service by querying the expected functional and nonfunctional requirements of available services. Selection is the process of picking the service that fits best under the current environmental parameters, thus optimizing the quality of the composed service workflow. The replacement is carried out by our component at runtime. These essential steps must be provided by our adaptive process component. The capability to adapt services in the business process dynamically at runtime is one of the main innovations in which this solution improves the resilience of critical infrastructure services.

A short description of the application scenario is provided to achieve a better understanding of the goals, challenges and environment of the SERSCIS workflow components.

[1]`http://www.serscis.eu`

The scenario takes place on the airport and is dedicated to the turn-round process. This process starts when the aircraft arrives "on block" and ends when the aircraft vacates the parking position. The controlling actor within this process is the ground handler. He communicates with other parties (e.g.: catering company, fuel company, airline) and consumes other services (e.g.: catering service, refueling service) in order to keep up with his tasks. The activities for the turn-round process from the ground handler's point of view are shown in Figure 1.

2. ADAPTIVITY OF BPEL PROCESSES

In SERSCIS, our workflow description language is BPEL. The system architecture is designed to be open for other workflow languages as long as a suitable execution component is available. However, the selection of a stable workflow representation language for this work is an important step. We chose BPEL plus semantic annotations for following reasons: BPEL contains a well defined mechanism for treatment of failure situations, which is of utmost importance in critical infrastructures. Besides, BPEL has gained the status of a standard in service orchestration, resulting in a rich repository of mature tools available for handling BPEL workflows.

In this work, we treat three important issues of BPEL when using a critical infrastructure environment. First, BPEL has no built-in support for dynamic adaptation. If a business process is encoded in BPEL and deployed into a respective execution environment afterwards, the workflow cannot - within the BPEL engine - easily be adapted dynamically at runtime taking security and dependability restrictions into account. Second, service endpoints or references to other services cannot be changed without editing and redeploying the deployment-description. Third, the BPEL standard has no built-in support for monitoring a running process. Nevertheless, monitoring of quality service parameters is a necessary requirement to decide whether a replacement of a service should be performed or not.

The chosen execution environment of the testbed is built around the *Apache ServiceMix* [3] enterprise service bus (ESB) and the *Apache ODE* BPEL execution engine [2]. From the deployment perspective of the execution engine, the following artifacts form a deployable workflow, namely a BPEL file, one or more WSDL files and a deployment descriptor for the ESB.

2.1 Dynamic Service Binding

In a critical infrastructure ICT system environment, it is important to select appropriate services that satisfy all security restrictions. There is still a need for dynamic service matching. This function introduces more flexibility in the choice of services while maintaining controllable behaviour of the system. Service classes, the relations between them and related security levels are modelled in an ontology. This ontology is used for the service selection process and the risk management process [7]. The service descriptions with links to the ontology are stored in a service registry that can be queried.

Having partner links bound to concrete services is required for an executable workflow instance (also named a "process"). Being able to do this dynamically is a main source for system resilience. Binding services adds additional state information to a workflow instance (that is which concrete service is bound to which partner link). We assume that partner links can be identified by a name that is unique inside a workflow template and that service invocations using the same partner link are intended to be processed by the same service instance.

2.2 Service Matching and Selection

In order to find a set of suitable services for a given task the following steps are performed during service matching: A query is sent to the service registry to retrieve descriptions of services that belong to a class of services that is required for the affected partner link according to the semantic annotation in the workflow template. Based on the ontology it is possible to find all service classes that are compatible. More complex matchmaking algorithms are described in [4]. Further, only services that implement the required interface according to the interface definition in the WSDL file that describes the interface type associated with the affected partner link are selected. It is also checked that the service is usable under the Service Level Agreements. Finally, it is ensured that the non-functional characteristics of the service as defined by such an SLA can be used. In addition, it is compliant with the applicable workflow management policies. This results in a set of suitable services where the most appropriate service has to be selected according to an optimization goal defined for the affected partner link, on workflow level or by some general optimization policy.

There are several stages in the life cycle of a workflow instance that are eligible for service binding. Services can be bound when the workflow template is instantiated. The event of instantiation of a workflow can be used to search for a suitable service for each partner link in the workflow and for binding them right away. It is also possible to select the service when a message is addressed to an unbound partner link. This is late binding and is at the other end of the spectrum compared to the previous option. When the process is stalled due to a faulty resource (service) it is also necessary to re-select an appropriate service.

3. PERFORMANCE EVALUATION

The methodologies presented above enable adaptive BPEL processes to adapt to changing circumstances and make it possible to dynamically select and change services at runtime. In a standard BPEL process execution environment, it is not possible to exchange services at runtime which can lead to a halt of the entire process if a previously selected service becomes unavailable, even though another suitable service might be available. In the present approach, it is possible to ensure process execution also in adverse conditions as long as an eligible service is available. The advantage of this adaptivity is extremely valuable from an operational point of view as the occurrence of expensive disruptions can be minimized. In order to evaluate the overhead introduced by the adaptivity, we measured the execution time of a typical airport turn-round process as it occurs in the SERSCIS scenario and compared the dynamic and static service binding approaches. It has to be noted again that a static service binding is not able to adapt to changes in the process or the availability of services. In the case of the dynamic service binding, the registry needs to be queried for suitable services and stateful dependencies as explained in Sect. 2.1 need to be taken into account. The performance measurements have been conducted in a test environment and

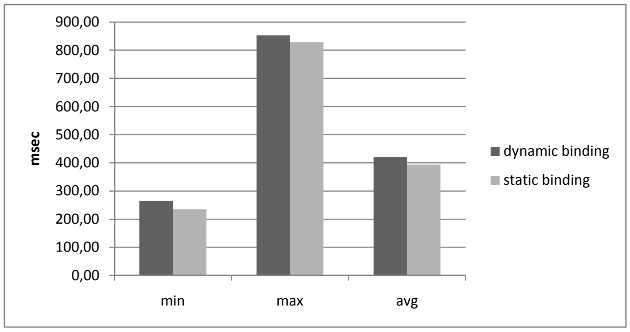

Figure 2: Runtime performance comparison between dynamic and static service binding

the performance for both (dynamic and static) cases will be considerably better in a production system. The execution times of the entire process using dynamic and static service binding are shown in Figure 2 with the minimum, maximum, and average execution times in milliseconds that were measured in 100 independent runs. On average, there is only a performance overhead of 22 milliseconds for the adaptive process with dynamic service selection compared to a static process execution. This increase of approximately 5% in execution time is an almost neglectable trade-off that will have no negative implications on the entire system's performance and thus it is recommendable to follow the adaptive process approach.

4. RELATED WORK

Attempts for making BPEL processes more adaptive have already been presented in various publications, but have shown restrictions in either flexibility of service descriptions including service selection and replacement or without considering security related aspects.

VieDame ([9]) performs BPEL scenario adaptation by using QoS parameters. These parameters and the resulting selection strategies are predetermined and implemented through pluggable models. These extensions are only available for ActiveBPEL engine [1]. In addition, this QoS do not consider any kind of security restrictions required in business processes for a critical infrastructure.

In [6] a framework for adaptive service selections was shown. It considers QoS criteria for adaptive BPEL processes and an exception resolution mechanism. To make it short, the limitation is the required pre-processing step and the QoS criteria cannot be changed afterwards. Thereby, this solution is not applicable in a CI.

Recent work only improving exception handling can be found in [5]. In their work, they consider also QoS parameters in the service replacement step; but it is not possible to change the replacement policy by the service consumer. Thus, this approach cannot be applied in CI.

Finally, entirely semantic dynamic service selection based on OWL-S, SA-WSDL, or WSML allows greater flexibility at the cost of precision and recall in the service selection (cf. [8]). In the last semantic service selection contest the best solution in terms of recall/precision had still an average response time of over 4 seconds for a set of a few hundred services. Such solutions are not applicable in a CI.

5. CONCLUSIONS

We have presented a methodology for making BPEL processes in service oriented architectures adaptive using semantically enhanced workflow descriptions. This approach enables late dynamic service binding and takes restrictions into account that are necessary to ensure security and resilience in critical infrastructure ICT systems. An evaluation of the system's performance has shown that the overhead introduced by the dynamic service selection is minor given that the impact of adaptability on the entire process execution is extremely positive. Our future work will focus on further enhancing the system performance and stability, improving user interface support, and evaluating the solution under adverse conditions to prove the applicability of the system for long-term use in critical infrastructures.

6. ACKNOWLEDGEMENTS

The research leading to these results has received funding from the European Community's Seventh Framework Programme under grant agreement n° 225336, SERSCIS.

7. REFERENCES

[1] ActiveBPEL. ActiveBPEL Community Edition. `http://sourceforge.net/projects/activebpel/`.

[2] Apache ODE. Apache ODE 2.0. `http://ode.apache.org//`.

[3] Apache Software Foundation. Apache ServiceMix. `http://servicemix.apache.org`.

[4] M. Baldoni, C. Baroglio, A. Martelli, V. Patti, and C. Schifanella. Service selection by choreography-driven matching. In *In Proc. of the 2nd ECOWS Workshop WEWST 2007, volume 313 of CEUR*, pages 1–17, 2008.

[5] K. Christos, C. Vassilakis, E. Rouvas, and P. Georgiadis. Exception resolution for bpel processes: a middleware-based framework and performance evaluation. In *iiWAS '08: Proceedings of the 10th International Conference on Information Integration and Web-based Applications & Services*, pages 248–256, New York, NY, USA, 2008. ACM.

[6] K. Christos, C. Vassilakis, E. Rouvas, and P. Georgiadis. QoS-driven adaptation of BPEL scenario execution. In *ICWS '09: Proceedings of the 2009 IEEE International Conference on Web Services*, pages 271–278, Washington, DC, USA, 2009. IEEE Computer Society.

[7] HMG. Information assurance maturity model and assessment framework. `http://www.cesg.gov.uk/products_services/iacs/iamm/media/iamm-assessment-framework.pdf`, 2009.

[8] M. Klusch, A. Leger, D. Martin, M. Paolucci, A. Bernstein, and U. Küster. 3rd international semantic service selection contest - performance evaluation of semantic service matchmakers. `http://www-ags.dfki.uni-sb.de/~klusch/s3/s3-2009-summary.pdf`, 2009.

[9] O. Moser, F. Rosenberg, and S. Dustdar. VieDAME - flexible and robust BPEL processes through monitoring and adaptation. In *ICSE Companion '08: Companion of the 30th international conference on Software engineering*, pages 917–918, New York, NY, USA, 2008. ACM.