International Academic Institute for Science and Technology

International Academic Journal of Science and Engineering
Vol. 5, No. 1, 2018, pp. 163-169.
ISSN 2454-3896

International Academic Journal of Science and Engineering

www.iaiest.com

Virtualization and encapsulation dynamic e-business services in a service-oriented architecture

Kianoosh Sadeghi[a]

[a] *Department of computer ravansar branch,Islamic azad university, Ravansar, Iran .*

Abstract

Service Provisioning over the internet using web service specifications becomes more and more difficult as real business requirements start to shape the community. One of the most important aspects relates to dynamic service provisioning: whilst the straight forward web service usage would aim at exposing individual resources according to a fixed description, real organizations would want to expose a flexible description of their complexly aggregated products. This paper presents an approach towards reducing the technological overhead in virtual service exposition over the internet, thus allowing for more flexibility. It therefore introduces a dynamic gateway structure that acts as virtual endpoint to message transactions and can encapsulate complex business process on behalf of the provider.

Keywords: business communications, encapsulating services, service-oriented architecture, gateway, Web Services

Received:24January 2018/Accepted:30 April 2018/Published:1 June 2018

Introduction:

Today's e-Business scenarios require a consequent realization of the Service Oriented Architecture (SOA) paradigm. Such a consequent realization provides benefits for both sides, the service providers as well as for the service consumers. Service provider can easily provide their "products" in such a way that potential service consumers can integrate these services in their own products. This is done in an abstract manner which means in particular that no implementation details of the underlying service implementation need to be considered. Service virtualization goes even one step further. Here operational, integration and life cycle issues are faced which is critical regarding the success of SOA. Service virtualization has already taken place in our everyday life. An example for such a virtual service is a banking service providing functionality allowing a client to execute financial transactions. Therefore in the background several underlying services are needed, like a transaction manager and a database system. The user of the banking service does not recognize these underlying subsystems since he only sees the interface of the banking service. Via this interface the complexity of the underlying infrastructure is hidden from the current user.

In modern e-Business scenarios it is necessary to decouple service implementations and the corresponding service interfaces. The main reasons therefore are that such a decoupling increases fundamentally the maintainability of services as well as the flexibility of both, service providers and service consumers.

In this paper we provide an approach towards virtual services allowing a decoupling of service implementations from the corresponding service interfaces.

Related work:

In [1] The modeling and performance analysis software system has done to the security aspects, the author of the analysis and design of formal policy-oriented analysis and design aspects of the non-operational features for real-time and distributed systems in the covers. [2] modeling and prediction performance realist architecture level includes in fact the author of this article has become a model to model And feature a special performance framework in the form of a firmware using existing analysis tools and techniques have been integrated simulation-based prediction. In [3] Fidel cached and colleagues analyzed data recovery performance distributed architecture using network simulation model was improved. In fact, they have several improvements over the two main bottlenecks in the network and server information retrieval system distributed by the network switch was performed. They also have shown that switch network between the internal network saturation, especially in the repeated decrease. [4] concepts, standards and solutions to provide Web services. In [5-10] are all examples of Web services is the implementation of standards.

e-Business and Web Services:

In current e-Business scenarios an abstract integration of collaboration partners is one of the main issues to be faced. In particular this means that partners within a collaboration want to consume the provided "product" of a partner without taking into account the corresponding service infrastructures. Web Services provide a first step towards such an approach. Web service technologies allow the consumption of services without the need to take into account the underlying service implementation. This is done by providing a standardized interface of these services (WSDL). These interfaces are integrated in the customers' code allowing him to consume the corresponding services. This interface just describes the functionality of the service in a syntactical manner. To announce a "product" consisting of the composition of several services enforces a more abstract view of the underlying services.

The main goals from an e-Business perspective are

• The easy encapsulation and usage of services being distributed all over the world

Received:24January 2018/Accepted:30 April 2018/Published:1 June 2018

• The easy composition of services in order to provide a "new product"

To realize these goals a new kind of infrastructure is needed with the goal to ease the maintenance of the underlying service infrastructures. In particular, changes of an interface or the service infrastructure should not affect the corresponding client applications. Additionally, service provider should also be able to easily adapt their infrastructures without affecting the corresponding interfaces and consequently the client applications consuming these services. The approach being presented in the following section is also going to ease the provision of new products regarding the currently available services.

A Dynamic WS Interface:

Currently WSDLs describe a static interoperable interface to a service which is used in static manner. The interface is once proposed and linked in a static manner in the corresponding client code. This static approach does not provide the needed flexibility in a dynamic e-Business scenario. To provide such an adaptive and dynamic infrastructure just a contract should be proposed describing the name of this "virtual" service as well as the available operations and what they mean in particular. Additionally it should be mentioned how these operations can be invoked.

Service virtualization provides such an infrastructure by not directly proposing a static interface in the means of WSDL, instead a kind of contract is proposed describing the available functionality and how these services can be invoked as well as which information is needed to invoke these services. The introduced middleware maps in the next step after having intercepted an invocation of such a virtual service endpoint the calls to the corresponding service implementations. The next sections are going to reflect this new approach in detail.

The New Gateway Architecture:

As mentioned before there is a concrete need in service virtualization and so consequently in an abstraction layer. This abstraction layer operates as an intermediary service between the service consumer and the service implementation by capturing the corresponding messages and mapping them to the corresponding services. This mapping also includes the necessary transformations since the virtualization gateway does not focus on a specific interface description.

Beside the mapping of messages to the corresponding service implementations within the service virtualization layer the following jobs can also be realized within this layer since the gateway describes a single point of entry to use the underlying services. This is preferable since most of the SOA infrastructures are some kind of "grown" nature with the restriction that some already existing implementations may not be compatible with current standards in interface definitions and messaging. So the gateway also provides functionality to encapsulate services. In particular, this includes:

• **Policy enforcement:** The gateway acts as a policy enforcement point since it allows the definition of criteria that must be fulfilled before a potential service consumer is authorized to access a specific service

• **Message security:** Identity and access management: In an ideal world, all deployed client applications and web services support the corresponding specifications like WS-Security, WS-Trust and WS-Federation. Ideally, each client application should be able to fetch security tokens that are necessary for service access, and each deployed service should be able to authorize an incoming request using a claims-based security model with fine-grained authorization.

• **Protocol translation:** Since standards in the area of web services are always a matter of change, the reflection of current needs of service consumers as well as of service provider are an essential criterion for such an infrastructure. In particular, the change of an addressing standard like WS-Addressing forces the adaption of the service implementations at the service provider side as well as the corresponding client applications consuming these services.

• **Transformation:** Since the gateway provides a universal interface for the underlying services a transformation has to be done before the message is forwarded to the corresponding service.

Received:24January 2018/Accepted:30 April 2018/Published:1 June 2018

• **Load balancing & fail over:** The gateway can act as a load balancer. If e.g. one service is currently heavy in use the gateway may decide to forward requests to this service to an equivalent one.
• **Routing:** If several equivalent services are available the routing of the messages to these services can be handled in this abstraction layer.
• **Login monitoring:** Often it is interesting for a service provider to see which version of a service is still used by the customers. Via the gateway this information is also available.
Figure I shows the structure of such a gateway. This structure enables service provider to encapsulate and hide their infrastructure in a way that also allows for virtualization of products. With the gateway being extensible, it provides the basis to non-invasively enact security, privacy and business policies related to message transactions. With the strong SOA approach pursued by the virtualization gateway, the structure furthermore meets the requirements of minimal impact and maximum deployment flexibility; through its filters, it furthermore supports the standardized messaging support. The gateway is furthermore constructed in a way that allows for participation in multiple collaborations at the same time without requiring reconfiguration of the underlying infrastructure.

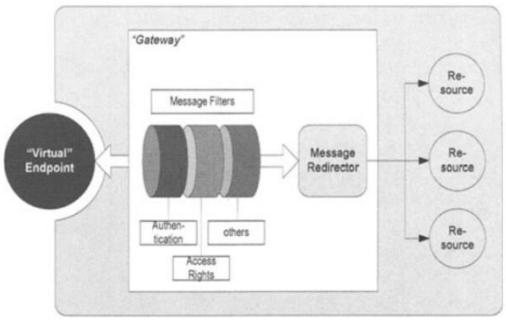

Fig. 1. Gateway Structure

The gateway of a service provider acts as the virtualization endpoint of the services exposed by the respective organization. Its main task consists in intercepting incoming and outgoing messages to enforce a series of policies related to access right restrictions, secure authentication etc. (cp. Figure 2) thus ensuring that both provider and collaboration specific policies are maintained in transactions.

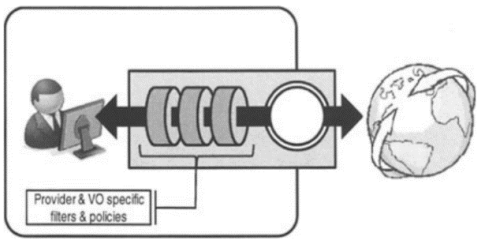

Fig. 2.The Gateway Principle
Figure 3 shows the conceptual overview of such an approach. In particular, the virtualization manager of a service provider announces a virtual service interface definition (WSDL). This virtual interface is also announced by the web server of the service provider to receive external service calls via the included

Received:24January 2018/Accepted:30 April 2018/Published:1 June 2018

virtual methods. These calls to the virtual interface are forwarded to the virtualization manager. In the following proceeding the virtualization manager transforms the incoming virtual message to a message that can be interpreted by the corresponding service implementation. Therefore the virtualization manager accesses a knowledge base containing all the necessary information like e.g. the mapping of the virtual name to a concrete service endpoint and the transformation of method names and parameters.

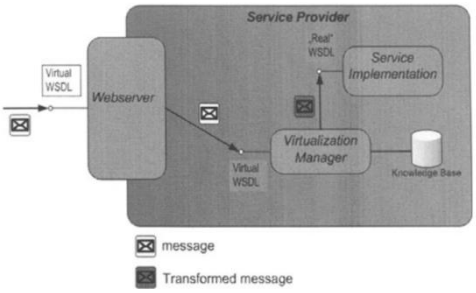

Fig. 3. General Architecture

To provide a service virtualization manager, an ideally transparent intermediary service is needed acting as a message interceptor und as a message transformer. In particular, in the area of web services a HTTP router is needed doing this transformation without affecting the client calling the corresponding service as well as the under-lying service implementation. Figure 4 illustrates an example of this processing:

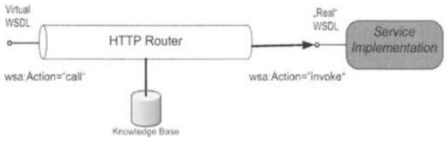

Fig. 4. Technical Realization

Trust Management:

In distributed-system scenarios, the main security problem is cross-organizational authorization. Most identity and access systems available today provide flexible solutions for authorization-related problems within the boundaries of a single organization. Still, IT professionals who need security solutions for cross-organizational collaboration typically need to develop their own custom solutions.

E-Business and the New Gateway:

This approach introduces a new abstraction layer for SOAs facing the needs of e-Business environments. In particular the main benefit is an increment of flexibility: Both, for the technical as well as for the business perspective, flexibility has been increased. From a technical point of view it is now possible to bind services statically in application codes while the corresponding service implementation can be migrated. Additionally the service provider can announce the available services independently from the protocol the potential service consumer are going to use. This way of announcing services allows the service provider to use and re-use already existing services in a very easy way. Beside this, the composition of services in a workflow has also been improved: Depending on the target outcome of a workflow services can now be combined regarding the announced contract. The service provider is

Received:24January 2018/Accepted:30 April 2018/Published:1 June 2018

consequently able to provide "new" products depending on the currently available resources, services and their current payload.

Resulting from this increase of flexibility, the main benefits of this approach are:

• Increased customers satisfaction: service providers are now able to adapt very fast to different customers' needs.

• Easy and improved maintenance of provided services

• Efficient development since the customers' technical point of view does not need to be considered within a concrete service implementation.

• Easy adaptation of provided services to changing web standards. Since web standards in the area of security, addressing, reliable message transfer, etc. are continuously under development and improvement , the corresponding service provider has to support as most of these standards as possible.

• Decreased costs

• Loose coupling can be better realized with such an approach

• Monitoring and logging in abstraction layer: enables the administrator to see which versions of a specific service are mostly used

• Governance guidelines force the realization of specific functionality which is often not conforming with the current service realization. The presented approach can realize this requirement without affecting the service implementation.

• Service consumer may use different end user systems to consume the corresponding services

• Many "grown" SOA infrastructures available are already existing and need to be integrated. This can be realized with an extremely reduced effort with the presented approach.

Conclusions:

In this paper we presented an approach towards a "real" SOA paradigm and how this can only be realized with a corresponding support of a service virtualization infrastructure. We also presented a conceptual approach to realize this service virtualization taking into account the already existing, partly grown SOA realization with web service technologies. Finally we presented how this concept can be realized in principle taking into account the most common used web services infrastructures. The latter presentation showed that the current available concepts of these web service infrastructure implementations allows an adaptation of the "intelligence" of a service virtualization infrastructure in the sense that the corresponding knowledge support can be added in such a way that incoming messages of a virtual service definition can be mapped to a concrete service implementations.

References:

A.Sahai ,S.Graupner," Web Services in the Enterprise Concepts, Standards, Solutions, and Management", springer, New York,2005.

B.Verheecke, M. A.Cibrán, V. Jonckers," Aspect-Oriented Programming for Dynamic Web Service Monitoring and Selection", European Conference ECOWS, Germany, 2004, 15-29.

C. Pahl," A Conceptual Framework for Semantic Web Services Development and Deployment", European Conference ECOWS, Germany, 2004, 270-284.

K.Sachs, S.Kouneva,J.Baconb, A.Buchmanna,Performance evaluation of message-oriented middleware using the SPECjms2007 benchmark,proc.66thPerformance Evaluation,Elsever,2009,410-434.

M.Herr, U. Bath, A.Koschel," Implementation of a Service Oriented Architecture at Deutsche Post MAIL", European Conference ECOWS, Germany, 2004, 227-238.

Received:24January 2018/Accepted:30 April 2018/Published:1 June 2018

M.Marchi, A. Mileo, A. Provetti," Specification and Execution of Declarative Policies for Grid Service Selection", European Conference ECOWS, Germany, 2004, 102-115.

M.Vasko, S.Dustdar," An Analysis of Web Services Workflow Patterns in Collaxa", European Conference ECOWS, Germany, 2004, 1-14.

N. Y.Topaloglu, R.Capilla," Modeling the Variability of Web Services from a Pattern Point of View", European Conference ECOWS, Germany, 2004, 128-138.

P.Zhang , H.Muccini , B. Li, A classification and comparison of model checking software architecture techniques,proc. 83thThe Journal of Systems and Software,Elsever, 2010, 723–744.

ZH.L.Zhanga, K. F.Sheng, Automatic Conversion from UML to CPN for Software Performance Evaluation, proc.29thproceda engineering, Elsever, 2012, 2682-2686

Received:24January 2018/Accepted:30 April 2018/Published:1 June 2018