Semantic interoperability and transformation in enterprise integration

Abstract

The main purpose of enterprise interoperability try to cope with changeable business and variety of technologies by means of component identification, design, interconnectivity and service transformation. A rough roadmap initiated by consolidating all heterogeneous environment and system silo into component based foundation, and apply some particular methodology, framework and design to implement their common practices, the transformation requires different technologies and vision to achieve service virtualization which is a new area in future service industry to form a common main route, the service grid concept will provision a transparent and highly integrated framework to provide global service infrastructure which has no limitation in geography or system we use.

Introduction

Service transformation follow a common framework with common service and component for service consumers to share common business component, service component, common information schema and service bus in a transparent environment. Common component and enterprise interoperability are pre-requirements in this transformation and the reasons are;

i) In current technology, there is no other method to achieve ‘common practice’ unless componentization foundation is solid. This foundation offer flexibility in implement service component as each atomic element has their clear ‘boundary’ between each other, once they are identified and specified, component micro design is feasible and component interconnectivity can be easily defined.

ii) We define micro interoperability as: “component internal design from a macro architecture view, and macro architecture design from micro interconnectivity view”, this means the metadata not only carries information, they play semantic mediation role in both micro and macro design.

iii) In the service transformation, we need to prevent ‘big bang’ as most of the industry is not ready to take the risk from a fundamental change. We need to create intermediate steps for the enterprise to transform their business and technology. Most of the roadmap is bottom-up approach in consolidating infrastructure, application and business component shown in Figure 1.

As aggregated services and common components are critical in forming the enterprise skeleton, the design and aggregation process must be cautious from an interconnectivity perspective. This implies that high integrity, optimization, grain size, dimension in service provisioning, visibility, traceability, performance and other factors are areas which must be brought into consideration. An example is in the credit card business in which aggregated services have been built between the card holder and card issuer; in this case the same aggregation is also being formed between the merchant and the bank. The mediation service of an inter-banking fraud detector component has been designed by reading fraud profiles in real-time transaction environments to mediate the individual services into a business main route; this highly enhances the service provisioning in a particular dimension.

Literature review

It is becoming generally accepted in the information community that interoperability is one of the most important principles in metadata implementation. Other basic metadata principles include simplicity, modularity, reusability, and extensibility (Duval et al., 2002; Zeng et al., 2003). These principles inform metadata database design as well as other system-dependent developments. From the very beginning of a metadata project, the principles that enable user-centered and interoperable services should be foremost in design and implementation. Semantic interoperability goes beyond attempts to homogenize disparities through standards (Harvey 1999). A central question about semantic interoperability is how people and social groups with different perspectives identify and possibly resolve their semantic differences (Harvey 1999). The construction of information communities in heterogeneous institutional and disciplinary environments calls for frameworks to conceptualize and articulate these semantic differences (Harvey 1999). Semantic interoperability requires means to resolve complex differences that lurk behind apparently consensual terminology and procedures (Harvey 1999). Interoperability is the ability to exchange and integrate meaningful information (Vckovski, 1998).
In recent literature, a great deal has been written about achieving interoperability among different metadata schemas. A methodological analysis of interoperability focusing on knowledge organization systems (KOS) was presented in a previous article by the authors (Zeng and Chan, 2004). This article analyzes some of the methods currently used to achieve interoperability in a broader context, that is, among different metadata schemas and applications.

Aggregated services and common components should not be tightly coupled; instead, their dependency and interconnectivity should be based on a parameter and metadata level and their design and interfacing specifications should be fully visible for future development. The reason for this is that the more significant role these skeleton elements play, the higher the complexity of interconnectivity, business criticality and access frequency, which can result in significant impact if a defect is detected too late. In many instances, a defect is a kind of binding between components instead of a coordinate from aggregation.

A service aggregation needs to be implemented in an empirical rule-base in the same way that a phrase needs words, grammar, semantics and a certain level of common recognition. It should not be allowed to be handled for individual needs differently or randomly; this is the reason it should be designed and maintained at the enterprise level as a single view of enterprise solution.

The micro process starts with the mechanism for how the component is designed; this involves taking the business semantic of the macro level with coherence of enterprise interoperability between each layer. Before the design of the service component, we take direction from the business component by using a top-down approach to derive the component design while at the same time streamlining the alignment between service components and physical components by using both top-down and bottom-up approaches. Most of the bottom-up approaches are a componentization process working from existing information components, business objects, codes, system metadata or database access protocols. Once we construct the alignment between business, service and information and have the service component design and interconnectivity completed, we actively work on service layer transformation. The transformation is shown in Figure 2.

The reason this step is crucial is that the goal of vertical enterprise integration is to create interoperability between layers from both micro and macro perspectives, with both business and information layers working as a foundation to support the formation of the service layer which provides the actual services. Once the service component has completed its design with alignment to business and information, the focus can be shifted to micro interference within the macro view. In our previous research on metadata interoperability we revealed, in relation to both political and rational models, how the components are derived through the validation process and how the strategy of component implementation achieves layer interoperability. In this paper we extend our research in component design and technical implementation for both internal and interface component factors. [1]

**Micro process in component transformation**

As the objective is to build rule-based enterprise aggregated services, we introduce the “state” concept to represent the intermediate “enterprise status” between transformation and post-transformation. The state concept in our micro process transformation is not merely dealing with a component or a service, but with the service layer and its coherence of enterprise interoperability of business and information. The transformation therefore requires an event to incur the service with the intelligence to either predict the next service or to verify that the called service is recognized. We use the state concept discussed above to mediate the business, service event and information in the “time” dimension.

![Figure 1. Service virtualization and bottom-up approach](image1)

![Figure 2. Project architecture benchmarking](image2)
The state of the service layer and the business-service-information coherence is changed once we trigger an event and the service is successfully executed. The alphabet read in automata is leveraged to service component trigger and execution while the state change is leveraged to transformation within the service layer.

When the service event $\Phi_i$ (i $\Rightarrow 1$) is read and triggers the service component, we express the state transformation as follows:

$$
\varepsilon^i : r_{i+1} \times \Phi_i \rightarrow r_i
$$

we further derive the transformation as follows:

$$
\varepsilon^{i+1} = (r_i)^{x} \times \Phi_{i+1} \rightarrow r_{i+1} = (r_{i+1} \times \Phi_1) \times \Phi_{i+1} \rightarrow r_{i+1}
$$

which is in equilibrium to $r_{i+1} \times (\Phi_i \times \Phi_{i+1}) \rightarrow r_{i+1}$, and derives $\varepsilon^{mn} : r_n \times (\Phi_{n+1} \times \Phi_{n+2} \ldots \times \Phi_m) \rightarrow r_m$ ($n < m$), we call this the “semantic event aggregator.”

This means, the event ‘aggregate’ service component by using semantic metadata. When we eliminate number of event, we already simplify and build a potential main route for a particular service and bypass intermediate steps by using components’ interconnectivity and service plan to transform service into a solid enterprise skeleton. In any component context a formal interface contract for the invocation syntax is important. The semantics issue (the meaning of parameters and so forth) has to be solved as well (domain modeling). This is crucial to any business-to-business (B2B) and dynamic invocation scenario. Such scenarios are cornerstones of the service component vision of being flexible and agile in response to the new business needs in a world of mergers and acquisitions, business transformation, globalization and other relevant factors.

The string $\Phi (r_n, r_m) = \Phi_{n+1} \times \Phi_{n+2} \ldots \times \Phi_m$ is a path from $r_n$ to $r_m$, we call the “service route” and $r_{i+1} \times (\Phi_i \times \Phi_{i+1}) \rightarrow r_{i+1}$ a “composite service” between $r_{i+1}$ and $r_{i+1}$, we use the symbol $\Phi_i$ to denote a composite service.

When frequency $F(\Phi_{i+1}) = F(\Phi_i \times \Phi_{i+1})$ is higher than threshold $\sigma$, we aggregate the concatenated alphabet (part of the string) $\Phi_i \times \Phi_{i+1}$ into $\Phi_i$. $i+1$, this implies that the original state $r_i$ between $r_{i+1}$ and $r_{i+1}$ will no longer exist for this composite service.

When the business architect checks the result from semantic event aggregator - $\varepsilon^m$ which is more an upper-layer use case scenario, service route $\Phi_i$ which is in the service layer and its frequency - $F(\Phi_i)$, if $F(\Phi_i)$ is much higher than threshold $\sigma$, then from the business perspective the event aggregator $\varepsilon^m$ either consists of many composite services or is business critical. This means the aggregation service is the key within business and is crucial in daily operation from service perspectives and that fine tuning is feasible from a multi-fine grained boundary to a coarse grained service.

We call this the “main route” of the enterprise skeleton; in other words this is the highway between A and B. Virtual coarse grained service eliminates the boundary and leverages many fine-grained components which can be easily configured by adding or modifying the component dynamically to achieve the broader coverage of main route services. For this reason main route is one of crucial services in enterprise integration. One example is in the Telecommunication industry that incorporates an open service account, an open billing account and provisioning options for a new customer. This constitutes a main route for “new customer service” coarse grained service, though it is aligned with fine-grained components to support customer preference, service criteria and billing options which are branches within this main route.

In service route $\varepsilon^{mn}: r_n \times (\Phi_{n+1} \times \Phi_{n+2} \ldots \times \Phi_m) \rightarrow r_m$, ($n < m$) if these exist in a composite service $\Phi_i$ that $r_i$ is omitted ($n < i < m$). Within this framework individual communication protocols, data manipulation languages and data are represented in a generic manner while preserving their ontological variety; this process indicates how a mediation infrastructure can be built using canonical wrappers, operational requirements of integration and layered architecture for component design [3].

**Micro interoperability in the component-based design**

The importance of determining the common component and service aggregation is in building up an enterprise skeleton of vertical interoperability between business, service and information. This task is crucial, as in the business domain analysis stage, the impact area and scope to business components are minimized in decision making. Additionally, the business component is deconstructed into service components and physical elements for top-down alignment while the common component is not simply a symbolic service but instead requires a bottom-up approach to work on common areas; this serves to fine tune the common components and interfaces from the perspective of enterprise integrity and performance. The service aggregation is also important as the top-down deconstruction followed by bottom-up composition forms the skeleton of vertical interoperability and potential component-based implementation of the service process.

The iteration through each use case will be tracked for its path and service statistics so that once we parse all use cases we should be able to see the frequency of use of each service as well as the path and dependency between services. Where there is a high frequency of services we use the term “common services” and focus on the potential for reuse and minimization of the integration interface. In the case of service paths in which several services have a common sequence and appear in
use cases a certain number of times past the threshold, 
the services can be aggregated to a composite service or 
potential process. This needs to be designed at an 
enterprise level so that each organization has the same 
concept of and specification for this common service. 
Service aggregation means a composite component 
consisting of at least two components that form a unique 
sequence of execution order with a common interface 
within their boundaries; this composite component will 
be treated as a coarse grained service in provisioning the 
upward aggregation of services. The pre-built foundation 
of K-B-S-C alignment increases portfolio provision 
agility for each strategic decision K, so that each K (the 
“active map” consisting of the selected business 
components from the business component model) gives a 
clear picture of which business components should be the 
focus in deriving service elements.

Once the vertical skeleton and service extension are 
built, the interoperability reflected from the business 
layer has the capability of integrating cross-organization 
and cross-geographical constraints, as fully transparent 
services become possible if each system or organization 
follows an industry standard by using metadata strategy 
and an industry framework. An example of this would be 
a case in which several key industry vendors have built 
up an industry framework for service providers in that 
particular industry to follow, starting from business 
domain analysis and working through to component 
deployment. Each service provider would thus benefit 
from the industry common services and the common 
interface, particularly those involving merger and acquire 
cases (B2B, B2C). Any business activity becomes easier 
when the various parties involved have a common 
understanding and a consensus on business terms, service 
definitions, interfaces and component specification, 
particularly if these become transparent and common 
practices. The important concept here is that once the 
metadata of business, service and information constructs 
an industry standard, the organization’s vertical skeleton 
has the opportunity to integrate across multiple 
organizations for an easier global practice; this is a long-
term industry objective which we call service 
virtualization in a fully transparent environment.

The foundation of this skeleton is achieving 
interconnectivity between service components by using a 
common component and an aggregated component. We 
harmonize develop the service virtualization on top of this 
interoperability skeleton, which we call enterprise 
vertical integration, in extending the integration across 
industry and geographical areas. This is the virtualization 
service in a horizontal direction; once the industries of 
different regions build up their semantic interoperability 
in three metadata repositories (business, service and 
information), they can synchronize their business, 
operation and strategy even if they have their local 
implementation extended from the interoperability 
skeleton. This is what we call service virtualization; we 
predict that virtualization of services will become a 
priority in industry as semantic interoperability is in high 
demand to solve the issues in both vertical 
interoperability and horizontal interconnectivity.

Conclusion

We introduced concept of event aggregator which 
means, the event aggregates service component by using 
semantic metadata. We found within service components, 
there is a referential integrity between components and 
the relationship can be modeled and quantified by using 
the concepts of aggregation and automata. The analysis is 
crucial to any business-to-business (B2B) and dynamic 
invocation scenario as such scenarios are cornerstones of 
the service component vision of being flexible and agile 
in response to the new business needs in a world of 
mergers and acquisitions, business transformation, 
globalization and other relevant factors. In 
transformation of service aggregation we explored the 
possibility of applying automata theory in collaboration 
with service aggregation. In enterprise service 
provisioning and virtualization, we constructed the 
skeleton of enterprise service with leverage to each layer 
and the interoperability of the layers.

Service virtualization is a process of leveraging 
enterprise integration strategy within a specific 
geographical region or across regions. This can be an 
industry standard scenario such as a credit card 
transaction in which the service, geography and systems 
are fully transparent to both cardholders and merchants. 
A full virtualization environment needs to be built on top 
of the foundation of component design and service 
transformation process in order to apply the vertical 
enterprise integration approach to this final stage which 
is the goal of our research work. The standardization of 
metadata technology and common services and 
components are all requirements in this stage.

Reference

1. Jie Lu, Guangquan Zhang & Fengjie Wu, (2005), 
“Web-based Multi-Criteria Group Decision Support 
System with Linguistic Term Processing Function”, 
IEEE Intelligent Informatics Bulletin June 2005 
Vol.5 No.1
2. O. Zimmermann, P. Krogdahl, C. Gee, (2004), 
“Elements of Service-Oriented Analysis and Design”, 
IBM Academy
Framework”, Distributed and Parallel Databases 
Journal 1999