Pragmatic Sensory Data Semantics With Service-Oriented Computing

Robin Singh Bhadoria, Indian Institute of Technology Indore, Gwalior, India
Narendra S. Chaudhari, Visvesvaraya National Institute of Technology, Nagpur, India

ABSTRACT
The importance of an architectural semantic for service-oriented computing starts with the characteristics of software systems that has been recognized with sharing and the utilization of resources. However, an architectural characteristic for service-oriented based applications depends on interaction patterns that utilizes a data format in communication. These patterns also help in establishing communication between service components. The goal of service-oriented computing methodologies could be achieved by adopting a logical separation of services from the actual mechanism of resource assignment and allotment. This shows that how services are reliable as interaction patterns between service provider and requester. It generates the binding patterns between these two and establishes communication between them. This article discusses the mathematical-based semantic model that presents an architectural view for systems that follows a service-oriented computing methodology. It also briefs describes about different execution states and routes, which addresses the service specifications in service-oriented computing.

KEYWORDS
Cloud Server, Middleware, Sensory Data, Service Execution, Service Interaction Patterns, Service Semantics, Service-Oriented Computing, Wireless Sensor Network

1. INTRODUCTION
Service-oriented computing is an emerging paradigm where services are platform-independent computational entities. These services could be published, delivered, executed and discarded based on the quality of service (QoS) requirement (Phillips, 2013). It also supports the distributed, interoperable, and integration of services on common and shared platform called enterprise service bus (ESB). This platform aims at developing a novel approach for extending the features of service-oriented architecture (SOA) that support combining the “off-the-shelf” component-based software engineering. “QoS” may refer to extending the meaning of functional as well as non-functional support to service properties, protocol and business logic. This is well explained with building template to quality assurance as discussed in thesis (Reinecke et al., 2010; Kyusakov et al., 2013). Quality assurance methods typically inferred the service execution patterns which helps in better understanding the

DOI: 10.4018/JOEUC.2019040102

Copyright © 2019, IGI Global. Copying or distributing in print or electronic forms without written permission of IGI Global is prohibited.
actual need for service integration in legacy application like sensor network. Whereas fundamental theories and methodologies associated with data acquisition for sensor network need to address issues associated with service invocation. Communication could be strengthened by well-define ‘link’ between correspondent ‘node’. Here, ‘node’ could be client or just sensor node which actually rise request for services and communication between them is governed by a ‘link’. The service execution ensures the quality of data which is being exchanged between multiple nodes with help of desired protocol (Fiadeiro et al., 2012).

The streaming of sensory data could be possible with both local as well as cloud server. Whenever, such data is stored at a repository through a local server, its access is limited with specific number of user. On the other hand, when such data is make available at cloud databases through global server, its access is vital and rapid through various distributed algorithms and techniques. This scenario is well depicted in Figure 1 in which sensory data is traffic through cloud infrastructure at global resources. This arrangement could be possible with different service modeling in SOA systems.

1.1. Service Modeling

The business conversation is important style in engagement of services between its clients. This could efficiently be organized through exchange of correlated messages (Myers et al., 2010; Klaus et al., 2015). For example, an insurance company client may request for status update to his/her insurance refund. This creates an ‘event’ that does the invocation and integration of multiple services. This service invocation could either commit the refund or cancel it on basis of certain predefined features.

Figure 1. Sensory data streaming over local and cloud server
E-Services Privacy: Needs, Approaches, Challenges, Models, and Dimensions
www.igi-global.com/chapter/services-privacy-needs-approaches-challenges/18176?camid=4v1a

Comparison of the Features of Some CoP Software
www.igi-global.com/chapter/comparison-features-some-cop-software/18172?camid=4v1a