Abstract

The concept of Service Oriented Architecture revolves around registering services as tasks. These tasks are accomplished collectively by various disparate components seamlessly connected to one another. The task of interlinking these components may be considered amongst the most convoluted and difficult tasks currently faced by software practitioners. This paper attempts to show that although middleware technologies can be solely utilized to develop service oriented architecture, however such architecture would severely lack quality, interoperability and ease of implementation. In order to resolve these complexities and complications this paper proposes Web Services as an alternative to Middleware, for the realization of a fully functional interoperable and an automated SOA which conforms to the characteristics of a SOA. This paper provides an abstract implementation model of a SOA using both middleware and web services. It then attempts to point out the implementation and accepted benefits of the latter, especially when legacy applications are involved. Emphasize is laid out on the significance of interoperability since it assists in mobility and other corporate benefits. The paper concludes that when interoperability along with its benefits of mobility, expansion, costs, simplicity and enterprise integration are required in the construction of a SOA then web services should be the definite integration choice. The paper also highlights the importance of object oriented middleware, along with situations in which it might be preferred over web services.
Comparing CORBA and Web-Services in view of a Service Oriented Architecture

References


Index Terms

Computer Science

Software Engineering

Keywords

Middleware CORBA Web Services SOA Interoperability