Chapter 5
Web Service Composition, Optimization and the Implications for Developing Economies

Adenike O Osofisan
University of Ibadan, Nigeria

Idongesit E. Eteng
University of Calabar, Nigeria

Iwara I Arikpo
University of Calabar, Nigeria

Abel Usoro
University of the West of Scotland, UK

ABSTRACT
The emergence of the Service Oriented computing paradigm with its implicit inclusion of web services has caused a precipitous revolution in software engineering, e-service compositions, and optimization of e-services. Web service composition requests are usually combined with end-to-end Quality of Service (QoS) requirements, which are specified in terms of non-functional properties e.g. response time, throughput, and price. This chapter describes what web services are; not just to the web but to the end users. The state of the art approaches for composing web services are briefly described and a novel game theoretic approach using genetic programming for composing web services in order to optimize service performance, bearing in mind the Quality of Service (QoS) of these web services, is presented. The implication of this approach to cloud computing and economic development of developing economies is discussed.

DOI: 10.4018/978-1-4666-1637-0.ch005
INTRODUCTION

According to Erl (2009), contemporary SOA (Service Oriented Architecture) represents an open, agile, expensive, federated and composable architecture comprised of autonomous, (quality-of-service) QoS-capable, vendor diverse, interoperable, discoverable, and potentially reusable services, implemented as web services. Service Oriented Architecture is therefore an architecture consisting of a collection of services. These services communicate with each other either by passing simple data or by involving in complex interactions. The services can be used within multiple separate systems from several business domains. Therefore, a deployed SOA-based architecture will provide a loosely-integrated suite of services that can be used within multiple business domains. This could be of great benefit to developing economies as will be shown later. SOA also defines how to integrate widely disparate applications for a world that is Web based and uses multiple implementation platforms; this also is a major benefit.

Service-oriented architectures have been implemented in older platforms. Such platforms include Distributed Component Object Model (DCOM) or Object Request Brokers (ORBs) based on the CORBA specification. DCOM is an extension of the Component Object Model (COM). DCOM was introduced in 1996 and is designed for use across multiple network transports, including Internet protocols such as HTTP (Hypertext Transfer Protocol). It works primarily with Microsoft Windows. CORBA (Common Object Request Broker Architecture), on the other hand, was developed under the auspices of the Object Management Group (OMG) as a middleware that allows for interoperability of programs across vendors, computers, programming languages, networks and operating systems.

The implications of a Service Oriented Architecture include the following:

A. Interoperability is a key goal and technological differences must not inhibit the performance of services.

B. Services and applications must make their capabilities publicly known so that other existing or new services can take advantage of their existence. This also implies that it must be possible to compose complex and varied services from simple ones. The composed services are known as composite services.

C. Clients must have the opportunity of selecting the optimal (best) services that must be composed to meet chosen Quality of Service (QoS) criteria.

The concept of service orientation as a design paradigm therefore stipulates the necessary foundation for simplifying the enormous convolution and integration of system development. This is made possible because service orientation takes advantage of the autonomy of existing services and developers do not need to bother about developing systems from the scratch. This has obvious economic benefits especially where funds are limited and systems development expertise is scarce.

Service Oriented Architecture can be better understood when web services are understood. A web service is a piece of software that can be defined, described and discovered by XML artifacts Ouzzani (2004). We define Web services simply as application components that communicate using open protocols and can be used by other applications. They are self-contained and self-describing applications that can be discovered, used and composed into complex services. Quality of service attributes can be used for selecting and composing appropriate web services.

According to Liu & Baras (2004), in the Service Oriented Community, Quality of Service (QoS) comprises all non-functional attributes of a service, ranging from performance-specific attributes to security and cost-related data. In general, QoS can be grouped into deterministic and non-deterministic attributes. Deterministic