

웹서비스 품질요소의 정형화된 정의

강윤희*, 윤성현*, 조광문**, 이성훈*

*백석대학교 정보통신학부

**목포대학교 전자상거래학과

yhkang@bu.ac.kr

Formal Definition of Web Services Quality Factor

Yun-Hee Kang*, Sung-Hyun Yun*, Kwang-Moon Cho**,
Seong-Hoon Lee*

*Dept. of Computer & Communication Eng, Baekseok University

**Dept. of Electronic Commerce, Mokpo National University

요 약

최근 웹서비스의 사용은 메시지 수준의 상호운용성 및 시스템 통합의 용이성으로 인해 빠르게 확장되고 있다. 웹서비스 기반 응용 또는 웹서비스는 다른 웹서비스를 선택하여 이용하게 된다. 이러한 웹서비스의 선택은 정적으로 이루어지므로 신규 웹서비스에 대한 생성 및 서비스에 대한 상태 변화가 지속적으로 이루어지는 환경에서 최적의 서비스 결정 및 선택은 전체 응용의 성능에 영향을 미치게 된다. 웹서비스 사용자 간의 신뢰성과 안정성을 제공하기 위해서는 품질 정보를 유지하기 위한 하부 구조에 대한 확장이 필요하며, 품질 정보가 추가된 웹서비스에 대한 정보에 대한 정의 및 수집이 이루어져야 한다. 이 논문에서는 웹서비스의 품질요소에 대한 정형화된 정의를 기술한다.

1. 서론

웹서비스의 운영환경에서는 네트워크 지연 및 제한된 대역폭에 따른 성능의 저하, 보안 공격에 의해 취약성, 웹서비스 및 응용의 결합, 네트워크 분리에 따른 결합에 따른 가용성 문제, 메시지 손실 및 중복 메시지에 따른 신뢰성의 제약점을 가질 수 있다 [6]. 이를 해결하기 위해서는 등록된 웹서비스의 서비스 비용(overall cost), 성능(performance), 신뢰성(reliability) 및 확장성(scalability) 등의 서비스 품질요소(service quality factor)에 대한 정의 및 확장이 요구된다.

웹서비스 기반 응용 또는 웹서비스는 다른 웹서비스를 선택하여 이용하게 된다. 이러한 웹서비스의 선택은 정적으로 이루어진다. 그러나 웹서비스의 활용 분야가 기업 응용, 과학 기술 계산 응용 등으로 확대됨에 따라 신규 웹서비스에 대한 생성 및 서비스에 대한 상태 변화가 지속적으로 이루어지는 환경에서 최적의 서비스 결정 및 선택은 전체 응용의 성

능에 영향을 미치게 된다. 그러나 현재 웹서비스의 운영 및 관리는 SOA(Service Oriented Architecture)를 기반으로 이루어지고는 있으나 웹서비스의 품질 요소를 반영하지는 못한다.

웹서비스의 서비스 품질(service quality)은 응용 구축 과정에서의 서비스의 발견(discovery), 선택(selection) 및 버전 관리(version management)에 직접적인 영향을 주게 된다.

이 논문에서는 이를 위해 웹서비스에 대한 품질요소를 정의한 후 이를 정형화하기 위한 내용을 기술한다.

이 논문의 구성은 다음과 같다. 2절에서는 웹서비스의 주요한 특징을 살펴보고 3절에서는 웹서비스의 품질요소 정의 및 확장된 웹서비스 아키텍처를 기술한다. 4절에서는 결론을 기술한다.

2. 관련 연구

웹서비스는 인터넷을 통해 응용간의 동적인 연결

을 위한 분산 컴포넌트(distributed component) 기술로서 복잡한 통합 시스템에 대한 의존도를 줄이는 대신 메시지 기반의 상호 연동을 통한 약 결합의 통합 시스템을 구성한다. 웹서비스는 Java RMI, CORBA와 DCOM과 같은 분산 컴퓨팅 기술을 웹 기술에 차용하여 필요한 서비스를 웹에서 동적으로 찾은 후 서비스를 요청하고 그 결과를 사용하는 URI 기반의 접근 가능 소프트웨어 에이전트(software agent)를 총칭한다[1-3].

웹 서비스는 구현된 상태로 클라이언트 응용 프로그램에 제공되는 것이 아니라, 기능을 클라이언트 응용 프로그램에게 제공한다. 따라서 웹 서비스는 임의의 프로그래밍 언어를 이용하여 임의의 플랫폼에서 구현할 수 있으며 어떤 클라이언트 응용 프로그램과도 호환성을 가질 수 있도록 한다. 이와 같은 호환성은 웹 서비스가 XML 기반의 메시지를 사용하기 때문에 얻을 수 있는 장점으로 XML은 텍스트 형식을 따르며 정수의 바이트 순서와 같은 낮은 수준의 상세 명세에 의해 발생할 수 있는 모호함이 없는 중립적인 형태를 갖는다. 이러한 특징으로 인해 XML은 임의의 플랫폼에서 응용간의 통신을 위해 활용할 때 장점을 가지며 웹 서비스 역시 인터넷 상의 응용을 통합할 수 있는 기반을 갖는다[6].

웹 서비스는 비즈니스 로직을 갖는 서비스 컴포넌트를 사용하기 위해 공개되어진 서비스 인터페이스를 사용하여 응용을 구성할 수 있도록 기능을 제공한다. 그러므로 서비스 컴포넌트는 확장될 수 있고 특화되고 상속될 수 있으나 웹 서비스를 활용하는 응용에서는 서비스 컴포넌트의 인터페이스 정보만을 사용하여 응용을 개발할 수 있으므로 서비스 컴포넌트와 이를 사용하는 응용간의 결합도를 줄일 수 있는 장점을 갖는다.

웹서비스는 SOAP을 사용한 메시지 기반 표준 및 WSDL을 사용한 서비스 정의 기술 명세를 적용하므로 높은 상호운용성(interoperability)을 제공한다 [2,5]. 웹서비스에서 SOAP은 XML 포맷을 사용하여 웹서비스의 메소드를 호출할 수 있는 기능을 제공한다. WSDL은 서비스의 연결방법, 서비스를 사용하기 위한 방법 및 서비스 결과의 형태를 기술한다. [2,4].

3. 웹서비스 품질 요소 정의 및 확장된 웹서비스 아키텍처 설계

3.1 웹서비스 품질요소 정의

QoS 측면에서 서비스 품질은 응용의 실행 성능에 영향을 주는 시스템 차원의 자원들을 예측하고 관리하는 수단이다. 네트워크 대역폭과 같은 유한한 컴퓨팅에 따라 자원을 사용하는 응용에서 자원에 대한 예측은 기업 내 응용을 운용하는데 주요한 역할을 수행한다. 이러한 예측을 위해 기업의 응용 소프트웨어의 모든 요소에 대한 완전한 제어권을 갖는 전통적인 방법 대신 웹서비스를 이용하는 경우에는 비기능 요구사항(non-functional requirement)에 따른 척도에 의한 완전한 서비스 품질을 보장하는 웹서비스만을 이용하도록 한다[7-8].

표 1은 웹 서비스의 웹서비스 기반 기업 응용 설계를 위한 QoS 측면에서 정의한 서비스의 품질 척도를 보인 것이다.

표 1. QoS 측면에서의 서비스의 품질 척도

품질 요소	QoS 척도
가용성	서비스가 즉시 사용가능하도록 준비되어진 확률 - 평균 결함간 시간 MTTF - 운영 시스템의 웹서비스 가용 시간
접근성	한 시점에서 서비스 요청메시지에 대해서 성공적으로 응답하는 요청수 (응답메시지/요청메시지)로서 웹서비스 운영환경의 부하에 대한 평가에 활용할 수 있음
성능	웹서비스 운영 시스템에서의 다음 하위 속성 - 단위 시간당 서비스의 처리율 - 서비스의 지연 시간
응답시간	사용자의 웹서비스 요청에서 응답까지의 소요시간으로 클라이언트에서의 경과 시간
확장성	장비 추가 후에 처리율 증가 비율 - 단위 시간당 처리율의 증가
가격	서비스 사용에 관련된 비용 - 서비스의 사용유형에 따른 비용 - 요청당 사용 비용 - 기간당 사용 비용
신뢰성	웹서비스 운영 전반에서의 안정성 - 시간당 결함 횟수 - 결함 처리, 회복시간 - 오류간 평균시간 MTBE - 응답편차
안전성	인증, 접근제어, 암호화 적용에 따른 보안 수준 - 기밀성 유지 수준(전송, 메시지, 서비스) - 사용자 인증 처리 방법 - 접근 제어 방법 - ID 관리 체제 적용

품질요소의 정형화를 위해 가용성, 접근성, 지연시간, 응답시간 및 신뢰성을 정의하였다 여기서는 다

음은 품질요소 가용성 및 신뢰성에 대한 정의를 한정하여 기술한다.

- Availability (s, o) : 웹서비스 s의 연산 o가 호출 될 때 사용할 수 있는 확률
 - Availability (s, 0, i) : 웹서비스 s의 연산 o가 현재 시점에서 정해진 시간 간격(interval) 동안 사용할 수 있는 확률
 - Availability (s, o, 8:00) : 웹서비스 s의 연산 o가 오전 8:00부터 자정까지 사용할 수 있는 확률
 - HighAvailability (s, o) : 웹서비스 s의 연산 o가 95% 이상 특정 시간 안에 호출할 수 있음
- HighAvailability(s, o) :-
(> Availability (s, o) , 95)

다음은 응답시간의 표준편차를 기반으로 신뢰성에 대한 정의를 기술한 것이다.

- Reliability (s, o) : 웹서비스 s의 연산 o가 호출 될 때의 응답시간의 표준 편차값이 특정범위에 포함될 확률
 - Reliability (s, 0, i) : 웹서비스 s의 연산 o가 현재 시점에서 정해진 시간 간격(interval)의 응답시간의 표준 편차값이 특정범위에 포함될 확률
 - Reliability (s, o, 8:00) : 웹서비스 s의 연산 o가 오전 8:00부터 자정까지 응답의 표준 편차값이 특정범위에 포함될 확률
 - HigReliability (s, o) : 웹서비스 s의 연산 o가 응답시간의 표준 편차값이 특정범위에 90% 이내 포함될 수 있음
- HighReliability(s, o) :-
(> Reliability (s, o), 90)
- MiddleReliability (s, o) : 웹서비스 s의 연산 o가 응답시간의 표준 편차값이 특정범위에 90% 보다 적고 70 이상에 포함될 수 있음
- MiddleReliability(s,o) :
(< Reliability (s, o), 90)^(> Reliability (s, o), 70))
- LowReliability (s, o) : 웹서비스 s의 연산 o가 응답시간의 표준 편차값이 특정범위에 70% 밖에 포함될 수 있음
- LowReliability(s,o):-
(< Reliability (s, o), 70)

그림 1과 그림 2는 웹서비스 요청에 따른 응답시간에 대한 응답시간과 표준 편차에 대한 빈도수를

보인 것으로 응답 시간의 85% 이상이 평균응답시간인 7초에 근접한 결과를 보이고 있으며 87%이상이 표준편차의 3초 이내의 값을 가진다. 또한 이를 통해 해당 웹서비스는 중간 수준의 신뢰성을 갖고 있음을 평가할 수 있다.

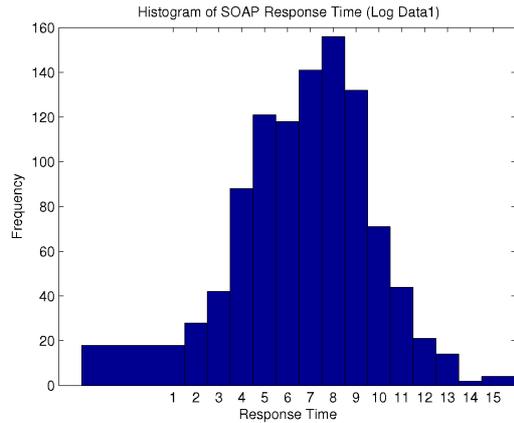


그림 1. SOAP 메시지 응답시간 별 빈도수

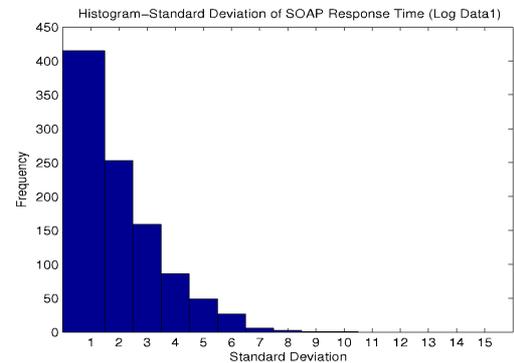


그림 2. SOAP 메시지 응답시간의 표준편차 별 빈도

3.2 확장된 웹서비스 아키텍처

웹서비스 품질요소의 관리를 위해서는 서비스의 등록 및 서비스의 사용에 대한 지속적인 모니터링을 위한 아키텍처의 관리가 필요하다. 이절에서는 앞 절에서 정의한 웹서비스 품질요소를 기반으로 웹서비스의 품질 정보 관리를 위한 확장된 웹서비스 아키텍처를 설계한다.

확장된 아키텍처에서 웹서비스의 품질정보는 응용 개발자에게 적합한 웹서비스를 찾도록 지원한다. 이를 위해서는 개별 웹서비스의 기술을 확장하여 품질정보를 유지하거나 관리체계를 통한 품질정보 수집 및 유지가 이루어져야 한다.

이를 위해 서비스 관리자를 구성하여 요청된 웹서비스에 대한 품질요소를 모니터링한 후 결과를 제공한다. 서비스 이용자인 응용은 서비스 관리자에

품질정보에 대한 비기능요건에 관련된 정보를 제공한다. 그림 3은 웹서비스 품질요소 관리를 위한 확장된 SOA 구조를 보인 것이다.

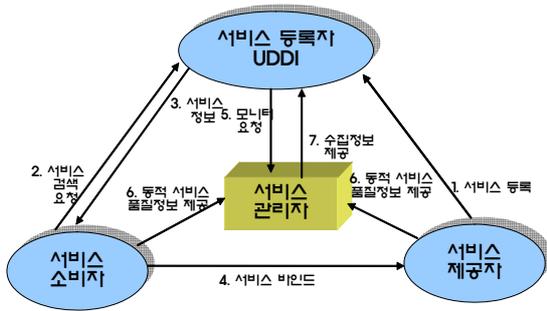


그림 3. 웹서비스 품질요소 관리를 위한 확장된 SOA 구조

4. 결론

공공부문 및 기업의 전산 시스템은 제어가 가능하여야 하나 웹서비스를 기반으로 응용을 구축하는 경우 서비스를 제공하는 제3 공급자에 대한 제어가 쉽지 않으므로, QoS을 기반으로 한 예측가능성을 제공하기 위해서는 웹서비스 품질에 대한 평가는 필수적으로 요구된다.

이 논문에서는 웹서비스에 대한 품질 모델을 기반으로 확장된 웹서비스 아키텍처 설계하였으며 웹서비스의 품질 정보를 품질 요소별로 정의하였다.

참고문헌

[1] M.P. Papazoglou and D. Georgakopoulos, "Service-Oriented Computing," CACM, Vol. 46, No. 10, Oct 2003.

[2] W3C, Web Services Architecture, <http://www.w3.org/TR/2003/WD-ws-arch-20030808/>

[3] F. Curbera et. al., "Unraveling the Web Services Web: An Introduction to SOAP, WSDL, and UDDI", IEEE Internet Computing, Vol. 6, No. 2, pp. 86-93, March/April 2002.

[4] W3C, Web Services Description Language(WSDL) 1.1, 2001, <http://www.w3c.org/TR/wsdl>

[5] W3C, Simple Object Access Protocol (SOAP)

1.1. 2000, <http://www.w3c.org/TR/SOAP>.

[6] P. Baglietto, M. Maresca, A. Parodi and N. Zingirian, "Deployment of Service Oriented Architecture for a Business Community," In Proc. of the Sixth International ENTERPRISE DISTRIBUTED OBJECT COMPUTING (EDOC'02), 2002.

[7] T. Erl. Service-Oriented Architecture (SOA): Concepts, Technology, and Design. Prentice Hall, 2005.

[8] 박진수 외, 엔터프라이즈 웹서비스 개발, 홍릉과학출판사, 2004