

# 복합 웹 서비스의 분산 처리 구조

박창섭, 이상수

KT 서비스개발연구소 플랫폼연구팀 플랫폼응용연구실

e-mail : [cspark0,sslle@kt.co.kr](mailto:cspark0,sslle@kt.co.kr)

## An Architecture for Distributed Processing of Composite Web Services

Chang-Sup Park, Sang-Soo Lee

Platform Research Team, Service Development Lab., KT Corp.

### 요 약

웹 서비스는 이질적인 응용 시스템들 사이의 연동 및 통합을 위한 표준화된 수단을 제공한다. 본 논문에서는 기존 웹 서비스들을 이용하여 정의되는 복합 웹 서비스를 효율적으로 실행하기 위한 방안으로서 사용자 에이전트를 이용한 분산 처리 시스템 구조 및 처리 방법을 제안한다. 본 시스템은 웹 서비스들의 통신 QoS 및 복합 웹 서비스의 부하 등을 고려하여 복합 웹 서비스의 호출 및 통합 작업을 사용자 에이전트에게 동적으로 위임하여 분산 처리함으로써 복합 웹 서비스의 성능 및 가용성을 향상시킨다.

### 1. 서론

최근 웹 서비스에 대한 관심과 적용이 산업계에서 점차 확산되고 있다. 웹 서비스는 XML 및 웹 프로토콜 기반의 표준화된 기술들을 통해, 분산되어 있는 이질적인 응용 시스템들 사이의 연동을 가능하게 한다. 이것은 웹 서비스의 기능 및 인터페이스를 어플리케이션 수준에서 인식 가능하도록 정의하는 웹 서비스 기술 언어(Web Service Description Language: WSDL), 그것을 외부에 공개하고 검색하기 위한 저장 시스템(Universal Description, Discovery, and Integration: UDDI), 그리고 다른 웹 서비스와의 통신을 위한 단순 객체 접근 프로토콜(Simple Object Access Protocol: SOAP) 등을 통해 이루어 진다[1]. 최근에는 이러한 기본 구조를 넘어서 웹 서비스들의 조합 및 통합, 서비스 품질 보장, 의미론적 서비스 기술 및 탐색 등을 위한 표준 언어와 프로토콜 명세들이 제안되고 있다[2]. 그림 1은 이러한 웹 서비스 요소 기술들을 계층화하여 나타낸 것이다. 웹 서비스는 궁극적으로 서비스 지향 컴퓨팅(Service-Oriented Computing) 패러다임을 실현할 수 있는 유용한 수단으로 발전하고 있다.

본 논문에서는 복합 웹 서비스(Composite Web Service)를 효율적으로 실행하기 위한 분산 처리 구조 및 방법을 제안한다. 복합 웹 서비스는 하나 이상의 요소(component) 웹 서비스들을 이용하여 구성되는 웹

서비스로서, 그 결과는 타 웹 서비스들에 대한 호출 결과를 기반으로 생성된다. 복합 웹 서비스는 중앙집중식으로 실행되므로 대량의 사용자 요구가 동시에 발생할 경우 처리 성능이 크게 저하될 수 있고, 요소 웹 서비스들이 모두 접근 가능하지 않으면 서비스 수행이 불가능하다는 문제점이 있다. 본 논문에서 제안하는 방식은 사용자 에이전트(agent)를 이용하여 복합 웹 서비스의 작업을 실시간으로 분산 처리함으로써 복합 웹 서비스의 성능 및 가용성을 향상시킨다.

본 논문의 구성은 다음과 같다. 제 2 절에서는 웹 서비스 복합화(composition)와 관련된 기존 연구들을 소개한다. 제 3 절에서는 에이전트 기술을 이용하여 복합 웹 서비스를 효율적으로 수행하는 분산 처리 구조를 제안하고, 제 4 절에서 세부적인 분산 처리 방법에 대해 기술한다. 제 5 절에서는 결론과 함께 향후 연구 방향에 대해 간략히 기술한다.

### 2. 관련 연구

일반적으로 웹 서비스의 내부 로직은 다양한 절차적 프로그래밍 언어를 이용하여 구현될 수 있다. 최근에는 기존 웹 서비스들을 이용하여 정의되는 복합 웹 서비스를 프로세스 모델을 이용하여 상위 레벨에서 효과적으로 기술하기 위한 언어들이 산업계에서 제안되거나 연구되고 있다. 이러한 언어로는 IBM 과

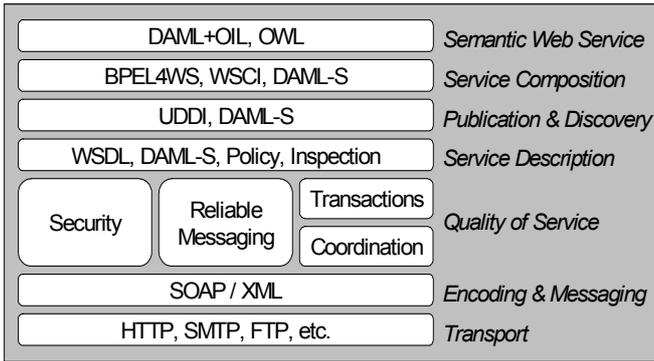


그림 1: 웹 서비스 기술 스택

Microsoft 등에서 제안한 BPEL4WS(Business Process Execution Language for Web Services), BEA 와 Sun 등에서 제안한 WSCI(Web Service Choreography Interface) 등이 있다[3]. 또한 최근 학계에서 활발히 연구되고 있는 시맨틱(semantic) 웹의 온톨로지(ontology) 개념에 기반하여 정의된 DAML-S(Darpa Agent Markup Language-S)는 웹 서비스의 동적 탐색 및 서비스 복합의 자동화를 실현할 수 있는 토대를 제공한다[4]. 한편 B. Benatallah 등은 상태도(statechart) 등을 이용하여 독립적이고 이질적인 웹 서비스들을 선언적(declarative)으로 조합하여 새로운 복합 웹 서비스를 정의하는 방법을 제시하였다[5].

S. Abiteboul 등은 웹 서비스 등의 외부 프로그램 호출을 포함하는 동적 XML 문서 구조와 그것의 분산 및 중복 기법을 제안하였다[6]. 본 논문에서는 사용자 에이전트에 대한 복합 웹 서비스 작업 위임의 수단으로 이와 같은 개념의 동적 XML 데이터를 이용한다.

L. Zeng 등은 웹 서비스에 대한 서비스 품질을 평가하기 위한 일반적인 모델을 제시하고 이를 이용하여 웹 서비스 복합을 위한 최적의 요소 웹 서비스들을 선택하기 위한 알고리즘을 제안하였다[7]. A. Mani 와 A. Nagarajan 은 다양한 웹 서비스 QoS 요구사항과 이를 충족시키기 위한 방안들을 기술하였다[8].

3. 시스템 구조

본 논문에서 제안하는 복합 웹 서비스 분산 처리 시스템의 전체적인 구성은 그림 2와 같다.

복합 웹 서비스는 네트워크 상에 존재하는 타 웹 서비스들을 이용하여 새로운 서비스를 구성한다. 일반적으로 웹 서비스는 사용자의 다양한 웹 클라이언트를 통해 접근 및 호출될 수 있다. 그러나 이런 클라이언트들은 호출된 복합 웹 서비스가 WSDL 에 의해 정의된 대로 실행 완료되어 결과를 반환하기를 기대하며, 컴퓨팅 능력이나 성능, 보안 등의 제약으로 인해 자체적인 서비스 실행 또한 일반적으로 불가능하다. 따라서 본 시스템에서는 사용자 클라이언트를 대신하여 작업을 자율적으로 수행할 수 있는 사용자 에이전트를 이용하여 복합 웹 서비스의 분산 처리를 수행한다[9]. 사용자 에이전트는 일반적으로 네트워크 상에서 사용자 클라이언트와 근접한 단말 노드에 존재한

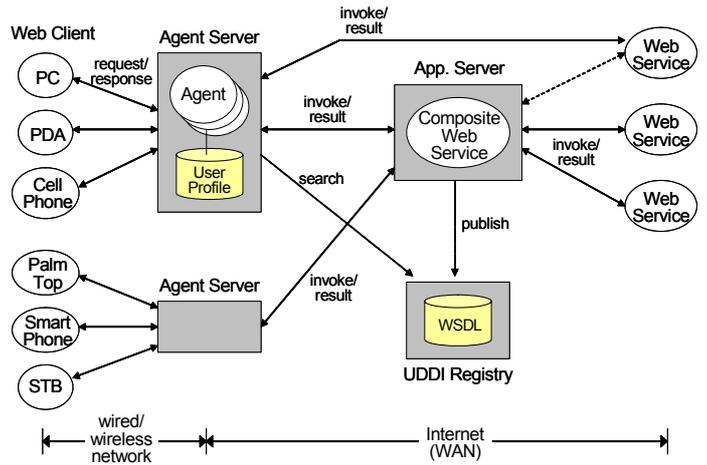


그림 2: 분산 처리 시스템 구성도

다. 즉, 사용자 클라이언트는 그것의 위치나 소속 등에 따라 특정 에이전트 서버에 유선 또는 무선망을 통하여 접속한다.

3.1 에이전트

본 시스템에서 에이전트는 특정 웹 서비스에 의존적이지 않고 독립적으로 실행되는 프로그램 객체로서, 사용자로부터 요구 사항을 전달 받아 사용자를 대신하여 적절한 복합 웹 서비스를 호출하고 그 결과를 사용자에게 전달한다. 이 과정에서 복합 웹 서비스 작업의 일부를 직접 수행할 수 있다. 에이전트 시스템의 내부 구조는 그림 3 에 도시된 바와 같으며, 각 구성 요소들의 기능은 다음과 같다.

- 컨텍스트(context) 관리기 및 사용자 프로파일(profile): 사용자 에이전트는 기 저장된 사용자 프로파일 및 문맥 정보를 이용하여 복합 웹 서비스에 필요한 사용자 정보를 전달함으로써 사용자를 대신하여 서비스를 개인화(personalization) 한다.
- 웹 서비스 검색기: 필요한 웹 서비스가 정적으로 바인딩(binding)되지 않은 경우, 사용자 요구 사항, 사용자 정보, 웹 서비스 의미 정보 등을 이용, UDDI 레지스트리(registry)를 탐색하여 최적의 웹 서비스를 찾는다.
- 웹 서비스 QoS 모니터(monitor): 특정 웹 서비스들에 대한 가용성, 접근성, 성능 등을 측정한다. 웹 서비스의 가용성(availability)은 웹 서비스가 호출되어 이용가능 할 확률을 의미하고, 접근성(accessibility)은 웹 서비스가 성공적으로 수행될 확률을 의미한다. 웹 서비스는 서버 및 네트워크의 오류(failure) 등에 의해 가용성과 접근성에 제약을 받는다. 웹 서비스의 성능은 웹 서비스 호출과 응답 사이의 지연 시간(latency)을 측정함으로써 간접적으로 비교될 수 있다. 이러한 QoS 값들의 계산은 웹 서비스 호출을 주기적으로 테스트하고 모니터링 함으로써 가능하다.

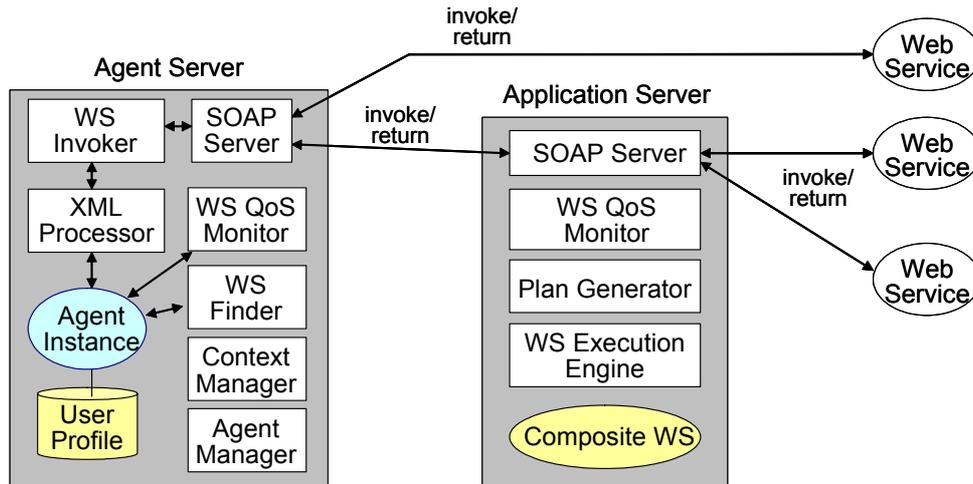


그림 3: 에이전트 및 복합 웹 서비스 시스템 구조

- 웹 서비스 호출기 및 XML 처리기: 복합 웹 서비스의 작업 일부를 수행하기 위해 필요한 요소 웹 서비스들을 호출하고 그 결과들을 통합하여 최종 결과를 XML 문서 형태로 생성한다.
- SOAP 서버: 웹 서비스 호출 및 반환 결과에 대한 SOAP 메시지의 인코딩/디코딩 처리를 수행한다.
- 에이전트 관리기: 사용자에 대한 에이전트 인스턴스의 생성, 바인딩, 영속화, 소멸 등 생명주기에 따른 관리 기능을 수행한다.

### 3.2 복합 웹 서비스

복합 웹 서비스를 수행할 응용 서버 시스템의 구조는 그림 3에 도시된 바와 같으며, 각 구성요소의 기능은 다음과 같다.

- SOAP 서버: 요소 웹 서비스 호출 및 반환 결과, 그리고 클라이언트/에이전트로 반환될 서비스 결과에 대한 SOAP 메시지 인코딩/디코딩 처리를 수행한다.
- 웹 서비스 QoS 모니터: 에이전트 내의 웹 서비스 QoS 모니터와 같이 요소 웹 서비스들에 대한 가용성, 접근성, 통신 비용 등을 측정한다. 그 외에, 복합 웹 서비스 서버 자체에 대한 부하 및 가용 자원을 모니터링 한다.
- 실행 계획 생성기: 에이전트의 웹 서비스 QoS 모니터를 통해 측정된 결과와 자체 웹 서비스 QoS 모니터링 결과를 비교 분석하여 내부적으로 수행할 작업과 에이전트에 위임할 부분을 결정한다.
- 웹 서비스 실행 엔진: 실행 계획 생성기로부터 결정된 내부 작업 프로세스를 실행하고 그 결과에 대한 XML 데이터를 생성한다.

### 4. 분산 처리 방법

일반적으로 웹 서비스의 내부 제어 흐름 및 비즈니스

스 로직은 불투명하게 구현되어 외부에 공개되지 않는다. 본 논문에서는 복합 웹 서비스의 제공자가 그것의 효율적인 분산 처리를 위해 사용자 에이전트가 필요로 하는 정보, 즉, 복합 웹 서비스를 구성하는 요소 웹 서비스들에 대한 주소 정보를 UDDI를 통해 공개한다고 가정한다. 또, 복합 웹 서비스의 내부 프로세스가 외부 환경 변수에 따라 조정될 수 있도록 유연하게 구현된다고 가정한다. 이러한 가정을 만족함으로써 사용자 에이전트를 이용한 복합 웹 서비스의 효율적인 분산 처리가 가능해진다.

본 논문에서 제안하는 복합 웹 서비스의 분산 처리 방식은 복합 웹 서비스에 포함된 요소 웹 서비스 호출을 사용자 에이전트로 위임하여 처리하는 것이다. 이러한 방식을 통하여 중앙집중식 복합 웹 서비스 서버의 부하를 사용자들의 서비스 요구량에 따라 적절히 조절하고, 네트워크 결합에 따른 요소 웹 서비스들의 가용성 저하 및 통신 비용 증가 등에 대처할 수 있다.

본 방식에서는 에이전트와 복합 웹 서비스 각각에 대한 요소 웹 서비스들의 QoS 및 복합 웹 서비스 부하 등에 따라 동적으로 복합 웹 서비스의 작업을 분할한다. 이를 위해 사용자 에이전트와 복합 웹 서비스 서버 사이에 부가적인 데이터 교환이 필요하다. 그러나 본 방식은 이러한 데이터 교환을 웹 서비스 호출 및 결과 반환 단계에 통합(piggyback)함으로써 분산 처리에 의한 네트워크 통신 비용의 증가를 최소화한다.

복합 웹 서비스의 분산 처리를 위한 세부 절차는 다음과 같다.

- 1) 에이전트는 요소 웹 서비스들의 URL을 포함한 복합 웹 서비스에 관한 정보를 UDDI 레지스트리로부터 얻는다. 웹 서비스 QoS 모니터는 요소 웹 서비스들의 URL 정보를 이용하여 그것들에 대한 연결성, 통신 비용 및 품질에 관한 정보를 수집한다.
- 2) 에이전트는 상기 QoS 측정값을 복합 웹 서비스

호출시 함께 서버에 전송한다.

- 3) 복합 웹 서비스 서버는 실행 계획 생성기를 통해 자체 수행할 작업과 사용자 에이전트로 위임할 부분을 결정한다. 이 때, 에이전트로부터 전달 받은 QoS 값과 자체 웹 서비스 QoS 모니터를 통해 측정된 통신 QoS 값 및 서버의 부하량이 결정 기준으로 이용된다. 서버의 부하가 일정한 임계값을 초과할 경우 가능한 많은 부분을 사용자 에이전트에게 위임하는 정책을 사용한다.
- 4) 양쪽의 작업이 분할되어 결정되면, 먼저 복합 웹 서비스 서버가 서버 측 프로세스를 웹 서비스 실행 엔진을 통해 실행한다.
- 5) 복합 웹 서비스 서버는 서버 측 수행 결과 및 에이전트로 위임할 웹 서비스 호출을 통합한 XML 데이터를 생성하여 에이전트에 반환한다.
- 6) 에이전트는 반환된 XML 데이터 내의 요소 웹 서비스 호출을 웹 서비스 호출기를 통해 수행하고, XML 처리기를 통해 그 결과와 기존 XML 결과를 통합한다.
- 7) 에이전트는 통합된 XML 데이터를 복합 웹 서비스에 대한 최종 결과로서 사용자 클라이언트에게 반환한다.

## 5. 결론 및 향후 연구 방향

본 논문에서는 기존의 웹 서비스들을 이용하여 정의되는 복합 웹 서비스를 효율적으로 실행하기 위한, 사용자 에이전트를 이용한 분산 처리 시스템 구조 및 처리 방법을 제안하였다. 제안된 시스템은 요소 웹 서비스들의 QoS 및 서버의 부하 등을 고려하여 복합 웹 서비스의 호출 및 통합 작업의 일부를 사용자 에이전트에게 위임하여 처리함으로써 복합 웹 서비스의 성능 및 가용성을 향상시킨다. 단일 복합 웹 서비스에 대한 다수의 사용자 요구에 대해 사용자 에이전트를 증가시킴으로써 peer-to-peer 방식의 분산 시스템과 같은 효과를 얻을 수 있다.

향후 연구로는 본 시스템의 성능 평가 및 개선, 시맨틱 웹 서비스를 이용한 웹 서비스 복합의 자동화에 관한 연구가 필요하다.

## 참고문헌

- [1] A. Tsalgatidou and T. Pilioura, "An Overview of Standards and Related Technology in Web services", *Distributed and Parallel Databases*, 12(2), pp.135-162, Sept. 2002.
- [2] F. Curbera, *et al.*, "The Next Step in Web Services", *Communications of the ACM*, 46(10), pp.29-34, Oct. 2003.
- [3] M. Solanki and C. Abela, "The Landscape of Markup Languages for Web Service Composition", Technical Report, De Montfort Univ., 2003.
- [4] S. A. McIlraith, *et al.*, "Semantic Web Services", *IEEE Intelligent Systems*, 16(2), pp.46-53, March 2001.
- [5] B. Benatallah, *et al.*, "Declarative Composition and Peer-to-peer Provisioning of Dynamic Web Services", In *Proc. of Int'l Conf. on Data Engineering*, pp.297-308, Feb. 2002.
- [6] S. Abiteboul, *et al.*, "Dynamic XML Documents with Distribution and Replication", In *Proc. of Int'l Conf. on Management Of Data*, pp.527-538, June 2003.
- [7] L. Zeng, *et al.*, "Quality Driven Web Services Composition", In *Proc. of Int'l Conf. on WWW*, pp.411-421, May 2003.
- [8] A. Mani and A. Nagarajan, "Understanding quality of service for Web services", IBM developerWorks, Jan. 2002.
- [9] N. Jennings and M. Wooldridge, "Software Agents", *IEE Review*, 42(1), pp.17-20, Jan. 1996.