

웹서비스를 이용한 네트워크 관리 시스템 설계

박노삼^o 이길행
한국전자통신연구원
{siru23^o, ghlee}@etri.re.kr

Design of Network Management System Using Web Services

Nosam Park^o, Gilhaeng Kim
ETRI

요 약

플랫폼과 프로그래밍 언어에 상관없이 애플리케이션을 이용할 수 있는 웹서비스가 차세대 플랫폼의 대안으로 부상하고 있다. 현재의 웹서비스는 XML 을 이용한 애플리케이션 통합에 중점을 두고 있다. 본 논문에서는 웹서비스를 이용한 네트워크 관리 시스템의 구조를 제안한다. 또한 다양한 플랫폼, 프로그래밍 언어를 이용하여 개발되어 있는 기존의 개별적인 네트워크 관리 시스템에 웹서비스를 도입하는 방안을 제시한다. 본 논문의 웹서비스 네트워크 관리 시스템은 에이전트를 이용하여 사용자가 요청한 웹서비스를 탐색, 전달해 준다.

1. 서론

웹서비스의 출현은 새로운 형태의 e 비즈니스 진화를 나타낸다. 웹서비스는 인터넷 기반의 모듈화된 애플리케이션으로서, URI(Uniform Resource Identifier)로 식별되는 소프트웨어 시스템을 나타낸다. 웹서비스는 시스템간의 연동을 지원하기 위해, 현재 브라우저를 통해 사용자와 애플리케이션 간의 상호작용을 위해 사용되는 HTML 을 초월하는 새로운 차원의 지능적 언어를 필요로 한다. 이에 따라 웹서비스의 인터페이스와 바인딩은 XML 로 정의 및 기술된다.

웹서비스 프레임워크는 통신 프로토콜인 SOAP(Simple Object Access Protocol), 서비스를 기술(description)하는 WSDL(Web Services Description Language), 서비스를 탐색(discovery)하는 UDDI(Universal Description, Discovery, and Integration)로 구분할 수 있다.

기존의 네트워크 관리 시스템은 네트워크 서비스 공급자가 제공하는 해당 서비스별로 별개의 시스템을 구축하였다. 서비스 공급자의 각 서비스별 네트워크 관리 시스템 또한 각기 다른 기술과 플랫폼을 이용하여 구축되어 통합적인 네트워크 관리 및 네트워크 관리 시스템간 상호 운용이 어려운 현실이다. 통합 네트워크 관리를 위해서 JMX(Java Management Extension), WBEM(Web-based Enterprise Management), CORBA 를 이용한 네트워크 관리 등 여러 방안들이 연구되었다.

본 논문에서는 플랫폼, 프로그래밍 언어에 상관없이 이용할 수 있는 웹서비스의 가장 큰 장점인 상호운용성을 보장하는 방안에 대해 기술하였다. 이를 바탕으로 웹서비스를 이용한 네트워크 관리 시스템의 구조를 제안하였다.

본 논문의 구성은 다음과 같다. 2 장에서는 웹서비스와 관련한 연구를 알아보고, 3 장에서 웹서비스를 이용한 네트워크 관리 시스템에 대해 살펴본다. 마지막으로 5 장에서 결론을 맺는다.

2. 관련 연구

웹서비스는 표준화된 XML 메시지를 통해 네트워크 상에서 접근 가능한 연산들의 집합을 기술하는 인터페이스로 정의된다[2]. 본 장에서는 현재의 웹서비스 기술에 대해 알아보고, 웹서비스의 향후 연구 방향인 시맨틱 웹서비스에 대해 살펴본다.

2.1 웹서비스 개요

웹 환경에서의 XML 기술을 이용한 분산 컴퓨팅을 이용함으로써 웹서비스는 웹기반의 시스템 통합을 용이하게 한다. 그에 따라, 웹서비스는 CORBA, Java RMI, DCOM 과 같은 기존의 분산 컴퓨팅 모델을 대체할 수 있는 차세대 기술로 부각되고 있다.

웹서비스는 특별히 새로운 기술은 아니다. ASP(Application Service Provider)는 수 년 동안 웹 기반의 서비스를 최종 사용자에게 제공해 왔다. 그러나 다양한 애플리케이션 간의 상호 탐색 및 데이터 교환을 인터넷상에서 유기적으로 가능하게 하는 것은 웹서비스이다. 예를 들어, Java 로 작성되어 Solaris 에서 동작 중인 프로그램이, 서비스가 어떻게 구현되어 있는지에 대해 고려할 필요없이 C#으로 작성되어 윈도우 XP 에서 동작하는 코드나, 리눅스 상의 Perl 프로그램을 실행할

수 있다.

웹서비스는 기존의 애플리케이션 지향적인 특성에서 벗어나, 사용자뿐만 아니라 애플리케이션간의 통신이 가능하다는 특징을 가진다. 그리고 서비스 레지스트리를 이용한 서비스의 검색 및 바인딩이 가능하다.

현재 웹서비스는 SOAP, WSDL, UDDI 에 대해 표준화를 진행하고 있는 상태이다. 주요 IT 업계들은 이 표준들을 기반으로 웹서비스 아키텍처를 설계 및 제안하고 있으며 최근에는 웹서비스 아키텍처에 대한 표준화를 W3C 에서 진행하고 있다[3].

웹서비스는 아직도 중요한 기능으로 사용되어야 할 여러 표준들이 정해지지 않은 상태이다[4]. 이러한 기능들 중 가장 중요한 것들이 보안, 트랜잭션 처리, 워크플로우 처리에 관한 표준이다. 이러한 기능들에 대한 표준화 작업은 계속 진행 중에 있다. 그 중 SAML(Security Assertions Markup Language)은 사용자 인증 정보 교환에 관한 프레임워크로 OASIS XML 보안 서비스 기술 위원회에서 표준화 작업을 진행하고 있다.

현재의 웹서비스는 실제 e-비즈니스 적용에 있어 필요한 기능을 충분히 제공하지 못하고 있다. 따라서 트랜잭션, 보안, 비즈니스 플로우 정의 등과 같은 여러 문제점을 해결해야 한다. 2001 년 8 월에 열린 Workshop on E-Business and the Intelligent Web 에서는 이러한 e-비즈니스를 지원하기 위한 새로운 종류의 웹서비스를 제안하게 되었다[5].

2.2 시맨틱 웹서비스

시맨틱 웹(Semantic Web)의 주요 목적은 웹서비스와 비슷하다. 웹을 사용자가 인지하여 이용하는 것이 아니라, 기계가 웹을 처리할 수 있도록 하는데 그 목적이 있다.

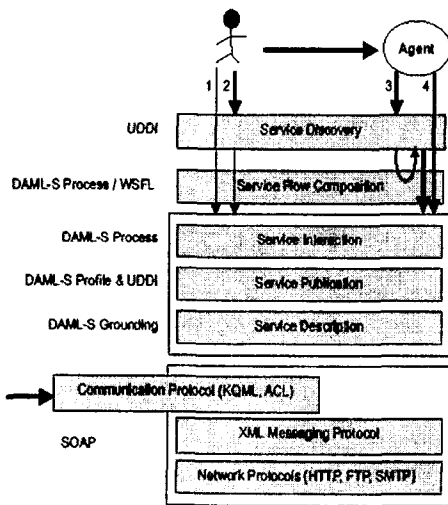


그림 1. 시맨틱 웹 서비스 아키텍처

■ 시맨틱 웹서비스 아키텍처

시맨틱 웹서비스의 아키텍처는 그림1과 같다[6]. 시맨틱 웹서비스는 UDDI에 해당하는 서비스 탐색(Discovery), 기술 기반의 상세 사항인 서비스 기술(Description), 비즈니스와 관련된 상세 사항인 서비스 배포(Publication), 서비스의 내부 처리 흐름의 상세 사항인 서비스 상호작용(Interaction) 등으로 구성되어 있다. 또한 HTTP, FTP, SMTP 등의 다양한 네트워크 프로토콜들에 대해서 XML 기반의 SOAP을 이용한다.

온톨로지(Ontology)는 반구조화된(semi-structured) 데이터와 제약조건들을 표현하는 기능과 타입과 상속을 지원하는 기능이 뛰어나다. 반면에 웹서비스의 표준들은 관리성, 확장성, 모놀화에서 우수하다. 따라서 DAML-S와 웹서비스 표준의 장점을 모두 이용하기 위해서는 두 가지 기술을 융합할 필요가 있다.

이와 같은 관점에서 다음과 같이 두 개의 시나리오로 나누어 생각할 수 있다

첫 번째 시나리오는 사용자가 웹서비스에 접근하는 경우이다. 이때 두 가지 가능성이 있는데, 사용자가 직접 웹서비스에 접근하는 경우 (그림 1의 1번 화살표)와 서비스 탐색 레이어에 있는 레지스트리를 이용하는 경우(그림 1의 2번 화살표)이다.

서비스를 직접 접근하는 것은 사용자가 서비스명과 위치 등을 알 때에만 가능한 방식으로 자주 사용되지는 않는 경우이다. 대부분은 서비스 탐색을 이용하여 서비스를 찾고, 서비스에 대한 명세를 이용한다.

두 번째 시나리오는 에이전트가 웹서비스를 찾아내어 사용자의 요구에 맞는 작업을 호출하는 시나리오이다. 이 경우, 에이전트가 직접 해당 서비스를 찾아 접근할 수도 있고, 에이전트가 레지스트리를 이용할 수도 있다.

시맨틱 웹서비스에서는 이와 같이 에이전트를 이용함으로써 서비스 호출을 좀더 효율적으로 지원할 수 있는데, 이는 기반이 되는 온톨로지가 확장성이 뛰어나기 때문이다.

3. 웹서비스를 이용한 네트워크 관리 시스템

표준화된 XML 기반의 인터페이스를 이용하는 웹서비스는 플랫폼과 독립적이고 프로그래밍 언어에 종립적이다. 이러한 방식으로 네트워크 상에서 응용프로그램들을 통합할 수 있으며, 분산 환경 하에서 애플리케이션들 간의 실시간 상호작용을 촉진시킨다.

기존에 개발되었던 개별 네트워크 관리 시스템은 통합적인 관점에 대해 고려 사항 없이 개별 서비스별로 네트워크 관리 시스템을 구축하였다. 네트워크 사업자 측면에서 개별 서비스의 네트워크 관리 뿐만 아니라 모든 서비스에 대한 네트워크 정보를 관리할 필요성이 제기되었다. 이에 따라 통합 네트워크 관리 시스템을 위해서 많은 연구가 이루어졌다. 본 장에서는, 기존 네트워크 관리 시스템의 변경을 최소로 하는 통합 네트워크 관리를 위해서 웹서비스를 이용한 구조를 제시한다.

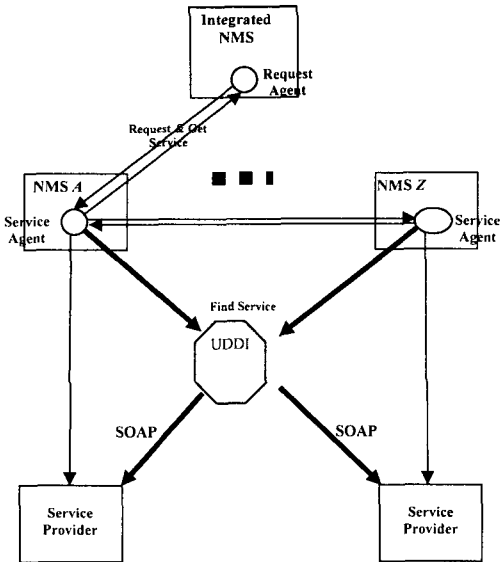


그림 2. 웹서비스 네트워크 관리 시스템의 구조

그림 2 는 웹서비스를 이용한 네트워크 관리 시스템의 구조를 나타낸 것으로서, 웹서비스의 기술인 UDDI, SOAP, WSDL 을 이용하였다. 통합 네트워크 관리를 위해서 필요한 정보는 에이전트를 이용해서 획득한다. 종합적으로, 본 논문에서 제시하는 시스템은 에이전트를 이용한 웹서비스를 이용한 프레임워크이다.

웹서비스 네트워크 관리 시스템은 웹서비스를 요청하는 통합 NMS 와 요청 에이전트(Request Agent), 웹서비스 요청을 받아서 해당 서비스를 탐색 및 실행하는 NMS 및 서비스 에이전트(Service Agent), 웹서비스를 제공하는 서비스 제공자(Service Provider) 등으로 구성되어 있다.

개별 네트워크에 대한 정보를 얻기 위해서 통합 NMS 는 요청 에이전트를 이용해 웹서비스를 이용한다. 요청 에이전트의 요청이 개별 NMS 에 전달되면 각 서비스 에이전트는 UDDI 를 이용해 서비스를 탐색한다. 서비스 탐색 후 에이전트는 해당 서비스 제공자에게 요청을 전달한다. 최종적으로 결과는 해당 NMS 가 수신하여 통합 NMS 에게 전달한다.

특정 NMS 의 서비스 에이전트가 해당 영역의 서비스를 탐색하여 원하는 서비스를 찾지 못하는 경우는 에이전트를 복제한다. 즉, 동일한 요청을 가진 에이전트를 다른 NMS 에게 복제하여 동일한 과정을 반복하도록 한다. 복제된 에이전트는 원하는 서비스를 찾을 경우 해당 NMS 를 통해 통합 NMS 에 결과를 전달한다.

웹서비스를 이용한 네트워크 관리 시스템을 통해, 상호 플랫폼과 프로그래밍 언어가 다른 NMS 를 추가적인 비용을 최소화하여 통합할 수 있다.

4. 결론

본 논문에서는 통합적인 관정에 대한 고려 사항 없이 개발되었던 개별 네트워크 관리 시스템들에 웹서비스를 이용한 구조를 제시하였다. 향후 웹서비스 프레임워크간의 표준화가 이루어져야 웹서비스의 주요 목표인 상호 운용성을 확보할 수 있을 것이다.

네트워크 관리 시스템에 웹서비스를 도입할 경우 신규 플랫폼 도입 등의 비용이 필요없고, 기존 개발되어 있는 시스템에 최소의 프로그래밍을 통해 통합 네트워크 관리 시스템을 구축할 수 있다.

5. 참고 문헌

- [1] C. C. Clark and J. B. Cain, *Error Correcting Coding for Digital Communications*, Plenum Press, 1981.
- [2] David Orchard, "Web Services Pitfalls", <http://www.xml.com/pub/a/2002/02/06/webservices.html>, XML.com, Feb, 2002.
- [3] 이경하, 이규철, "웹 서비스의 향후 발전 방향," 한국정보처리학회지, 제 9 권, 제 4 호, pp. 24-30, 2002년 7월
- [4] 김현희, 차석일, 송준홍, 성백호, 신동일, 신동규, "SAML 기반의 웹서비스 프레임워크 보안에 관한 연구," 한국정보처리학회 춘계학술발표대회, 제 9 권, 제 1 호, pp. 932-934, 2002년
- [5] Alun Preece and Stefan Decker, "Intelligent Web Services," *Intelligent Systems*, IEEE, pp.36-47, Jan/Feb 2002
- [6] Sollazzo T., Handschuh S., Staab S., Frank M., "Semantic Web Service Architecture - Evolving Web Service Standards toward the Semantic Web," *Proc. of the 15th International FLAIRS Conference*, Pensacola, Florida, May 16-18, 2002
- [7] Zdenek Svoboda, "Developing Interoperable Web Services - Integrating Java and Microsoft .NET," <http://www2.theserverside.com/resources/article.jsp?l=Systinet-web-services-part-5>