

● 목 차 ●

1. 서 론
2. XML Web services이 바로 Microsoft .NET
3. Microsoft .NET의 2기 그리고 구체화되는 XML Web Services
4. 결 론

1. 서 론

우리가 알고 있던 비즈니스는 이제 사라지고 있다. 다시 말해, 지출한 비용대비 수익을 고려하지 않은 채, 회원수와 클릭횟수에만 집중했던 닷컴시대의 비즈니스는 더 이상 존재하지 않는다. 수익성과 전통적인 비즈니스 필요조건들의 진리가 다시 한 번 입증되고 있으며, 이제는 웹 사이트를 구축하고 홍보를 하는 것만으로는 부족하다. 성장을 원하는 기업들은 비용을 절감하는 동시에 더 많은 일을 해야만 하며, 전통적으로 비용발생 부서인 IT 부서도 이제는 경쟁적 우위를 확보하는 것과 함께 수입을 발생시킬 것을 요구받고 있다. 이와 동시에, 비즈니스는 물리적인 공간에 제약을 받지 않으며, 직원들도 자신의 자리에서 하루 8시간 근무가 아닌 전세계와 접촉하며 24시간 일하고 있다. 사람들은 인터넷을 사용하여, 전세계의 협력업체, 공급업체, 고객과 접촉하고 있다.

비즈니스가 변화되면서, 이에 따른 기술도 변화하고 있다. 컴퓨팅 파워는 18개월마다 2배씩 증가하고 있으며, 이 추세는 향후 5년에서 10년까지 계

속될 것으로 예상된다. 그 결과는 누구나 예측할 수 있다. 예를 들면, 휴대폰은 5년 전의 고성능 PC와 맞먹는 처리능력을 가지게 될 것이며, 처리능력은 크게 향상되고 있는 동시에, 통신비용은 급속히 낮아지고 있다. 인터넷 백본에 여유가 많아지면서, 네트워크의 종점인 개인과 기업들에게도 백본수준의 대역폭이 제공되기 시작했다.

그리고, 휴대폰, PDA, 자동차와 같이 새로운 영역에서 인터넷이 사용되기 시작했다. 아침에 눈을 뜨면, 새로운 장비들이 웹을 활용하여 새로운 서비스를 제공하고 있다. 이러한 장비들을 유용하게 사용하려면 매우 많은 작업이 필요하지만, PC와 함께 정보를 생성하여 교환할 수 있는 강력한 수단이 되리라는 것은 의심할 여지가 없다.

중요한 정보를 별도의 시스템에 따로 유지하는 시대는 지났다. 오늘날과 같이 급변하는 비즈니스 환경에서는 여러 기업들이 함께 협업하던지, 아니면 고객 데이터베이스를 가진 웹 사이트에 접속하던지간에, 주요 시스템들이 서로 연동되게 하는 것이 생존을 보장받을 수 있는 방법이다.

기업 내부와, 그리고 비즈니스 협력업체와의 비즈니스 통합은 기존의 정보를 보다 잘 활용하고 중요하지 않은 비즈니스 기능을 외부에 아웃소싱하게 되

* 마이크로소프트 닷넷 팀 차장

어, 귀중한 시간과 자원을 절약할 수 있는 기회를 제공한다. 이러한 진리는 지불 기능을 아웃소싱하는 자전거 상점에서 수백개의 공급체계를 갖추고 있는 큰 자동차 제조업체에 이르기까지 모두 적용된다.

그렇지만, 통합이 쉬운 일은 아니다. 비즈니스 시스템을 통합하는데 소요되는 비용과 시간이 통합으로 인한 이점을 능가하는 경우가 많으며, 프로세스가 완료되어도, 비즈니스는 특정 시스템 또는 특정 협력업체에 한정되어, 또 다른 변환작업을 시작해야 하는 경우도 있다.

전통적인 통합비용은 여러가지 형태로 발생된다. 노동비 외에, 통합작업에 필요한 코드를 개발하고 유지보수하고 업그레이드하는데 귀중한 인적자원을 필요로 한다. 통합작업이 완료되면, 기업들은 특정 협력업체 또는 소프트웨어에 한정되어, 우선 순위나 비즈니스 환경이 변화하지 않기만을 바라고 된다. 기업이 계속 성장하려면, 시스템의 호환성 문제로 인한 제약을 받지 않으면서, 최상의 협력업체, 시스템과 애플리케이션을 신속하고도 유연하게 통합할 수 있어야만 한다. 기술과 비즈니스 환경은 계속 변화하기 때문에, 기업들은 경제적으로 통합되는 동시에 기술과 인력면에서 기존의 투자를 활용하는 신속성있고 다이내믹한 비즈니스 시스템을 개발할 수 있는 능력이 있어야 한다.

바로 이와 같은 상황을 반영하여 등장한 IT의 신조류가 XML Web Services이며, 이 신조류가 현실로 다가올 수 있도록 가장 적극적인 활동을 하고 있는 기업은 마이크로소프트다. Microsoft .NET은 XML Web Services를 마이크로소프트의 기술과 제품을 재구성하고 다른 기술과 통합될 수 있도록 하는 아키텍처이자 비전이다.

2. XML Web services이 바로 Microsoft .NET

XML Web service는 이와 같이 환경이 요구하는

통합을 가능하게 하는 새로운 기술이다. XML Web service는 기업 내에서 또는 외부 협력업체와 고객들과의 협업에서 컴퓨터, 애플리케이션, 휴대용 장비와 비즈니스 프로세스들이 서로 통신하고 데이터를 교환할 수 있게 한다.

연결을 위한 표준방법이 제공되면서, 더 신속하게, 더 쉽게 그리고 더 유연하게 통합하는 동시에 보안을 유지하고 비즈니스 요구사항을 해결할 수 있게 된다.

이것을 가능하게 만든 기초 기술은 XML (eXtensible Markup Language)이라는 프로그래밍 언어로, 애플리케이션 간의 데이터 교환을 위한 전문 언어다. 이 언어는 벤더 독립적이며, 플랫폼 독립적인 방법으로 애플리케이션들이 데이터를 서술하고 교환할 수 있게 한다. XML은 SOAP, WSDL, UDDI 등의 인터넷 표준과 결합하여, 소프트웨어 애플리케이션들이 개발된 시점에 관계없이 통합될 수 있게 한다.

.NET은 기업, 개발자와 개인 사용자가 XML Web Services를 보다 효과적인 방법으로 구현하고 사용할 수 있도록 하며, 정보, 장비와 사람들을 단일화되고 개인화된 방식으로 연결하는 차세대 소프트웨어 플랫폼이다. .NET 플랫폼은 XML 기반의 애플리케이션, 프로세스와 웹 사이트들이 정보와 기능을 플랫폼, 장비에 상관없이 서로간에 공유하고 결합하는 서비스로 사용하여, 기업과 개인의 요구사항에 적합한 솔루션을 제공할 수 있게 한다.

.NET 플랫폼은 XML과 인터넷 표준에 따라 디자인된 다양한 제품군으로 구성되어 있기 때문에, 개발(툴), 관리(서버), 활용(.NET Services와 스마트 클라이언트)과 사용자 경험(MSN과 같은 새로운 인터넷 경험)등 XML Web service의 각 부분을 제공한다.

2.1 기업을 위한 .NET의 비즈니스 통합

XML Web service를 위한 마이크로소프트의 플

랫폼인 Microsoft .NET은 기업들이 원하는 통합을 이룰 수 있도록 디자인되었기 때문에, 기업들이 내부 또는 외부적으로 비즈니스 협력업체, 고객들과 통합할 수 있게 한다.

XML Web service는 서로 다른 플랫폼 상의 애플리케이션과 정보들을 이전과 비교할 수 없을만큼 쉽고도 빠르게 연결할 수 있는 방법을 제공한다. 인사부서와 자금부서의 애플리케이션은 서로 연결되어 정보를 교환할 수 있기 때문에, 애플리케이션들이 중복된 정보를 유지하지 않고도 일관성있고 유용한 솔루션을 제공할 수 있다. 정보들이 각각의 애플리케이션에 추가되거나 업데이트되는 대신에 서로간에 소스의 정보를 직접 공유한다.

.NET으로 기업들은 내부 애플리케이션을 더 쉽게 통합할 수 있을 뿐만 아니라 다른 기업들이 제공하는 서비스를 활용할 수 있다. 기업들은 XML Web service를 사용하고 인터넷을 통해 연결하여, 자신의 시스템과 협력업체의 시스템을 연결하고, 매우 다양한 부가가치 애플리케이션을 개발할 수 있다. 예를 들면, 자동차 제조업체는 다양한 협력업체들이 제공하는 급여, 주식거래, 교육 등의 서비스를 하나의 통합된 직원용 포털로 만들어 직원들에게 서비스를 제공할 수도 있다. 또는 수많은 거래업체들의 재고관리, 물품보충 메커니즘, 구매주문 추적을 종합적인 공급선관리 시스템으로 통합할 수도 있다. 또한 기업들은 자신의 협력업체와 통합하여 고객들에게 더 완벽한 서비스를 제공하는 서비스를 추가할 수 있다.

다양한 소프트웨어 프로그램들이 연동될 수 있기 때문에, XML Web service는 기업들이 고객들에게 새로운 가능성을 제공할 수 있게 한다. 사용자들은 자신의 신원을 한 번만 확인하면 하나의 서비스에서 다른 기업의 서비스로 자연스럽게 이동할 수 있게 된다. 사용자들은 자신의 데이터를 통제하여 장소, 시간, 장비에 관계없이 활동할 수 있게 된다. 그들은 다른 사이트 또는 기업의 서비스를

이용할 경우에, 자신의 정보가 사용되는 방법과 시기에 대하여 결정할 수 있는 권한을 가지게 된다.

2.2 일반 사용자를 위한 .NET의 보안과 편의성 강화

Microsoft .NET의 가장 기본적인 철학 중 하나는 사용자가 자신의 개인 데이터를 통제하여, 기술이 사용자를 위하여 사용될 수 있게 해야 한다는 것이다. 사용자가 데이터를 통제하게 되면, 온라인 개인 정보에 대한 논쟁은 사라지게 된다. .NET은 사용자가 자신의 모든 개인정보를 통제하고 어떤 조건하에서 이 정보를 누구와 공유할 것인지를 결정할 수 있다는 가정에서 출발하기 때문에, 기업들이 얼마나 많은 개인정보를 임의대로 사용할 것인지에 대한 논쟁은 더 이상 존재하지 않는다. .NET의 기본 철학은, 기업의 애플리케이션이 사용자 정보의 사용처와 시기에 대하여 사용자의 동의없이 결정하는 현재의 패러다임과 큰 대조를 보이는 것이다.

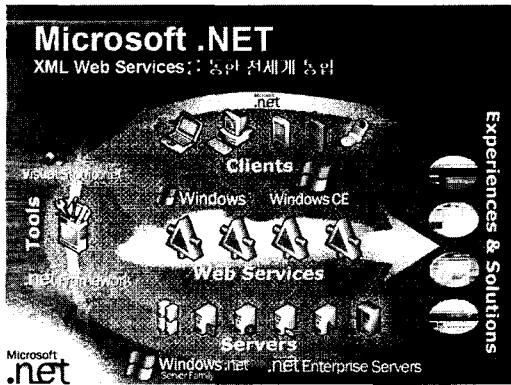
.NET은 특정 장비, 애플리케이션, 서비스 또는 네트워크에 종속되지 않고 사람을 중심으로 구현된다. 자신의 데이터를 자신이 통제할 수 있기 때문에, .NET은 애플리케이션, 장비와 서비스가 사용자를 위한 일을 할 수 있게 하며, 사용은 더욱 쉬워지고 개인의 선호를 더 중시하게 된다. 사용자는 .NET을 통해, 필요한 정보를 원하는 방식으로 원하는 장비에 전송되기 때문에, 자신이 사용하는 장비를 완벽하게 활용하게 된다.

많은 애플리케이션과 장비가 사용자가 통제하는 정보를 이용하기 때문에, 사용자는 서로 다른 기술 간에, 그리고 다른 사람과의 서비스 간에 정보를 안정적으로 공유할 수 있다. 현재와 같이 사용자가 기술에 적응해야 하고, 사용자가 직접 장비, 애플리케이션과 웹 사이트를 연결하는 역할을 하는 상황과 비교하면 분명한 차이가 있다. .NET을 통하여, 더 이상 한 서비스의 정보를 다른 서비스로 일일이 복사할 필요가 없다. 사용자는 더 이상 자신이 방

문했던 모든 곳에 자신의 정보를 갱신할 필요가 없으며, .NET은 사용자의 정보를 모아, 사용자가 사용하고 있는 장비를 통하여 유용한 형태로 제공한다.

2.3 .NET 플랫폼의 구성

.NET 플랫폼은 개발툴, 클라이언트, 서비스, 서버와 사용자 경험(Experience)의 5개 부분으로 구성된다.



(그림 1) .NET 플랫폼의 구성

2.3.1 개발툴

개발자들은 XML Web service를 구현할 수 있는 개발툴이 필요하다. 개발툴은 XML Web service를 구축하고, 전개하고 실행할 수 있는 환경을 제공한다. XML Web service를 사용하여 개발된 모든 애플리케이션과 서비스는 미래의 애플리케이션과 서비스를 위한 하나의 블록 서비스가 된다. 이것들은 XML과 같은 개방형 표준을 활용하기 때문에, 개발자들은 전체를 재구성하느라 귀중한 시간을 허비하지 않아도 된다.

마이크로소프트 개발툴은 XML Web service를 쉽고도 빠르게 개발할 수 있게 한다. Microsoft

Visual Studio.NET은 Microsoft .NET 플랫폼 상에서 애플리케이션을 구현할 수 있는 통합 프로그래밍 개발환경이며, .NET Framework는 .NET 플랫폼

의 중심에 있는 프로그래밍 인터페이스 집합으로, 개발자가 XML Web service의 성능, 안정성과 보안을 극대화시킬 수 있게 한다.

2.3.2 클라이언트

사용자들은 Web Services와 호환되는 장비와 소프트웨어를 통해 자신의 정보를 이용하게 된다. PC, 페이지, PDA, 휴대폰과 다른 스마트 장비가 사용되며, 소프트웨어는 장비들이 온라인 또는 오프라인 상태에서 다른 장비와 연결되어 사용자에게 더 풍부한 서비스를 제공할 수 있게 한다.

마이크로소프트는 현재 PC를 위한 Windows XP, 휴대용장비를 위한 Windows CE, 특수환경을 위한 Windows XP Embedded와 같이 다양한 스마트 장비용 운영체계를 제공하고 있다. 이러한 스마트 클라이언트 소프트웨어는 PC와 장비들이 온라인 또는 오프라인으로 사용자를 위하여 더 효과적으로 사용될 수 있게 도와준다.



(그림 2) .NET을 지원하는 스마트 장비

Microsoft .NET은 다양한 스마트 장비에 XML Web service를 적용하여 네트워크 상에서의 사용자의 경험을 더욱 풍부하게 한다. 사용자들이 사용할 수 있는 스마트 장비들로는 PC, 페이지, PDA, 휴대폰 등이 있다. 어떤 것을 스마트 장비라고 말하는 것일까? 스마트 장비는 다음과 같은 특징이 있다.

- **사용자를 잘 알고 있다:** 사용자의 .NET 신원, 프로파일, 선호사항 그리고 데이터를 사용하여 사용자 경험을 편하게 제공하며 사용자의 위치를 잘 알고 있어야 한다. 사용자가 어디에 있고, 어떤 장비를 사용하여 네트워크에 접속하고 있는지를 파악하여 사용자의 장비종류와 연결된 수준에 따라 전달하는 정보유형을 조정한다.
- **네트워크를 잘 알고 있다:** 대역폭한계에 따라, 사용자가 어떤 사용자 경험을 사용할 수 있는지를 파악한다. 애플리케이션을 온라인과 오프라인으로 모두 사용할 수 있게 지원하며, 사용자가 어떤 서비스를 이용할 수 있는지를 파악한다.
- **정보를 잘 알고 있다:** 사용자가 시간과 장소에 관계없이 데이터에 접속하여 분석한 후에, 그 결과에 따라 행동할 수 있게 한다.
- **다른 장비를 잘 알고 있다:** PC, 스마트 장비, 서버와 인터넷을 발견하여 연결하여 다른 장비에게 서비스를 제공한다. 그리고 PC에서 정보를 가져올 수 있다.
- **소프트웨어와 서비스를 잘 알고 있다.** 사용자가 사용하고 있는 장비에 최적화된 형태로 애플리케이션과 데이터를 제공하며, 적절한 형태의 입력과 연결방법을 제공한다. 그리고 XML, SOAP과 다른 웹 표준을 사용하여 웹 서비스를 제공한다.

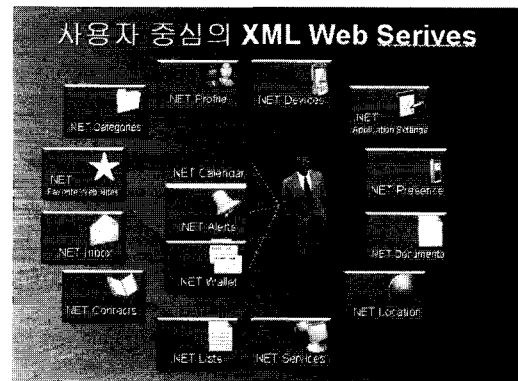
개발자들은 스마트 장비를 프로그래밍하고 확장할 수 있기 때문에, .NET Framework에 대한 지식을 .NET Compact Framework에 적용할 수 있다. .NET Compact Framework은 매우 작은 크기로 소프트웨어가 동작할 수 있게 한다. .NET Compact Framework은 스마트 장비들이 앞의 특징들을 모두 갖출 수 있게 한다.

2.3.3 서비스

XML Web service는 퍼즐조각과 같이 웹 상에 퍼져있는 애플리케이션들이다. XML Web service를 통해, 애플리케이션들은 데이터를 공유할 수 있을 뿐만 아니라 다른 애플리케이션이 XML을 지원하기만 한다면 다른 애플리케이션의 기능을 활용할 수 있다. XML을 통해 데이터를 공유하게 되면, 애플리케이션들은 서로간에 독립적인 작동을 하게 되며, 느슨하게 연결된 상태에서 필요에 따라 특정 작업을 함께 처리할 수도 있다.

마이크로소프트는 개발자들이 자신의 XML Web service를 만들 수 있도록 돕는 것 외에, 일반적인 작업을 수행하고, 개발자들이 활용할 수 있는 핵심 빌딩블록 서비스를 개발하여 제공하고 있다. 마이크로소프트는 .NET Services라는 첫 번째 XML Web service를 발표했다. 이것은 특정 장비, 네트워크 또는 애플리케이션에서 벗어나 사람이 중심이 된 사용자 중심의 서비스들이다. .NET Services를 통해, 사용자는 통지, 선호사항, 문서와 애플리케이션을 자신이 원하는 대로 설정하여 사용하거나 이미 설정한 내용에 따라 자신의 장비에 전달되게 한다.

“.NET Services”는 단절된 정보의 섬들을 쉽게 통합할 수 있는 사용자 중심의 아키텍처이자 XML Web service로, 특정 장비, 네트워크 또는 애플리케이션



(그림 3) .NET Services

이션이 아닌 사람을 중심으로 동작한다. 애플리케이션과 서비스들은 “.NET My Services”를 사용하여, 일반 사용자를 위한 협업을 하게 되며, 사용자, 그룹과 기업들은 서로 공유하고 협업할 수 있게 한다. 이런 서비스들은 가입하기만 하면 사용할 수 있기 때문에, 개발자들은 강력한 개발툴의 지원을 받아 잘 만들어진 솔루션을 제공할 수 있게 된다.

다음의 예는 “.NET Services” 빌딩블록들이 온라인 여행사의 항공편 예약과정을 얼마나 간단하게 완료할 수 있는지를 잘 보여주고 있다.

- 사용자의 명백한 동의 하에, 여행 서비스는 미리 저장된 사용자의 선호사항과 지불방법에 자동으로 접속한다.
- 사용자의 회사가 특정 여행규정을 가지고 있다면, 여행사는 그 회사의 “.NET Services” 그룹 신원에 접속한 후에, 사용자의 선호사항과 회사의 요구사항을 모두 만족시키는 일정만을 선택한다.
- 사용자가 항공편을 선택하면, 여행 서비스는 “.NET Services”를 사용하여 사용자의 캘린더 프로그램 종류를 파악하고, 사용자의 캘린더에 여행일정을 자동으로 갱신한다.
- 사용자의 일정이 변경되면, 여행사는 “.NET Services”를 통해 일정을 갱신하고, 사용자에게 스마트 장비를 통해 즉시 통보한다.
- 갱신된 항공편은 사용자가 정한 사람들에게 알려지고, 그 사람들은 사용자가 언제, 어디에 있는 것을 알 수 있다.

2.3.4 서버

XML Web service는 안정적이고 확장성이 높으며 전세계 모든 곳에서 사용될 수 있다는 면에서 엔터프라이즈 애플리케이션과 상당히 유사하다. 그러나, 현재의 애플리케이션 모델과는 달리, 모든 애플리케이션의 중심부에 한 대 또는 두 대의 대형 서버가 있지 않아도 된다. 대신에 XML Webservice

는, 요청을 인터넷 전체로 분산시켜 처리할 수 있다.

Microsoft Windows와 Microsoft .NET Enterprise Server는 XML 지원, 스케일 아웃(scale-out) 솔루션, 그리고 애플리케이션과 서비스 간의 비즈니스 프로세스 조율 기능을 포함하고 있기 때문에, XML Web service를 구축하고 전개하고 운영하기에 최상의 환경을 제공한다. 또한 절대 안정적인(Mission-critical) 성능을 염두에 두고 디자인되었기 때문에, 기업들이 XML Web Service를 통해 시스템, 애플리케이션, 협력업체와 통합할 수 있는 능력을 제공하며, 변화하는 비즈니스 요구사항에 적응할 수 있는 유연성을 제공한다.

Microsoft Windows 2000 Server와 .NET Enterprise Server는 미래의 XML Web service가 구현될 수 있는 기초가 된다. .NET Enterprise Server의 기능들이 결합하여, 기업들에게 소프트웨어가 서비스로 제공되고, 시간, 장소, 장비에 상관없이 정보를 이용할 수 있으며, 비즈니스 가치를 극대화시킬 수 있는 확장성 높은 솔루션을 제공한다.

.NET Enterprise Server는 대부분의 엔터프라이즈 애플리케이션을 위한 안정적인 플랫폼인 동시에, XML Web service를 전개하고 운영하고 관리할 수 있는 가장 효과적인 플랫폼이다. 이 서버들은 마이크로소프트가 향후의 신제품에도 계속 포함시킬 최신의 인터넷 표준을 지원한다.

2.3.5 사용자 경험(Experience)

사용자 관점에서의 결과물이 “사용자 경험”이다. 훌륭한 사용자 경험은 다양한 XML Web service, 애플리케이션과 스마트 장비가 결합된 결과이고, 한 기업에 의해 제공되는 XML Web service보다는 다양한 기업의 서비스와 결합하게 될 때, 완벽한 사용자 경험이 만들어진다.

마이크로소프트는 지식근로자, 소비자, 기업, 중소기업과 개발자들에게 사용자 경험을 제공할 것

이다. 마이크로소프트가 사용자 경험으로 전환하여 제공하는 제품들로는 Microsoft Office, MSN 인터넷 접속, Microsoft bCentral 중소기업용 포털 그리고 Visual Studio.NET이 있다. 이와 같은 .NET 사용자 경험은 특정 사용자가 요구하는 XML Web service와 클라이언트 소프트웨어를 하나로 연결하고, 사용자를 만족시키는 통합방식으로 사용자에게 모든 것을 제공할 것이다.

마이크로소프트는 현재 4개의 유명제품들을 .NET 경험으로 변경시키고 있다. Microsoft Office XP는 지식 근로자를 위하여 .NET 경험을 제공하는 첫 번째 제품이다. MSN은 MSN Explorer 로컬 클라이언트와 함께, 사용자 중심의 .NET 경험을 제공할 예정이다. Microsoft bCentral small business 포털은 중소기업들을 위한 재고관리와 같은 XML Web service를 제공하고 eBay와 같이 중요한 XML Web service를 사용하게 된다. Visual Studio 개발 시스템은 개발자들이 사용하는 툴안에서 직접 MSDN 정보와 해당 기업만의 코딩 가이드라인을 제공하여, 개발자를 위한 .NET 경험을 제공한다.

3. Microsoft .NET의 2기 그리고 구체화되는 XML Web Services

3.1 .NET의 통합 하부구조 GXA (Global XML Architecture)

3.1.1 GXA란?

잘 알려져 있다시피, XML Web Services의 핵심 표준인 SOAP 1.0은 Microsoft사의 COM+ 팀과 Don Box에 의해 정립되었다. SOAP외에 현재 공통적인 XML Web Services 표준으로 인정되는 것들로는 WSDL (Web Services Description Language), UDDI (Universal Description, Discovery and Integration) 등이 있다. 물론 이를 통해 P2P에서 필요한 기본적인 기능들을 구현하는 것이 가능하다. 그러나 현재의 XML Web Services 표준은 대부분 공통적인 통신

레이어에 대한 것들이다. 즉, 보다 풍부한 서비스 레이어의 표준들(Security, Routing, Discovery, Monitoring, Management,...)이 존재하지 않는다. 즉, 이러한 표준의 부재는 어플리케이션 개발자가 직접 정의해야 하거나, 이 기종간의 통신에 어려움을 겪을 수 있다는 것을 의미한다. GXA(Global XML Architecture)란 보다 광범위한 범위의 XML Web Services 표준을 작성하기 위한 프로토콜 프레임워크다. 즉, 제품이 아니라 표준의 집합이며 이는 P2P를 포함한 모든 XML Web Services 어플리케이션을 위한 서비스 레이어 표준 프로토콜들이다. 물론 GXA는 Microsoft사가 주도하는 표준의 집합이지만, GXA의 모든 표준들은 WS-I.org를 통해 검증 받고, 발표된다.

현재 GXA의 주요 표준으로는 Metadata & discovery, Messaging, Security, Transaction이 있으며 Security와 관련된 WS-Security 표준들은 전체 로드맵과 함께 IBM과 더불어 WS-I.org에 제안이 된 상태다. 기타 표준들 역시 조만간 출시될 것이다. 결국 .NET 프레임워크는 GXA의 주요 표준들을 포함하게 될 것이며 나아가 Microsoft 사의 기술과 제품들은 이러한 표준을 실제 구현함으로써 플랫폼 독립적인 상호 운영성을 보장하게 될 것이다. 예를 들어 .NET Passport의 경우 WS-Security를 지원함으로써 플랫폼 독립적으로 연합되어지며, 분산되어진 형태의 XML Web Services 인증 시스템 중 하나가 될 것이다. 물론 SUN Liberty Alliance가 WS-Security 표준을 지원한다면, 두 진영 간의 인증 매커니즘 역시 통합이 가능할 것이다. 다시 말해 두 WS-Security 표준으로 인해 더 이상의 .NET Passport와 SUN Liberty Alliance간의 도덕적인 전쟁은 의미를 잃게 되었다. 정리하자면, GXA가 P2P를 포함한 XML Web Services 어플리케이션들 간의 통신 프로토콜을 정의 한다면, .NET 프레임워크는 그 자체가 XML Web Services기반의 P2P 어플리케이션을 위해 풍부한 기능을 제공하는 미들웨어가 되

는 것이다.

| | |
|---------------------------|-------------------------------------|
| Peer Discovery Protocol | TCP/IP - Network protocol |
| Peer Resolver Protocol | HTTP - Transport protocol |
| Peer Information Protocol | XML - HTTP payload format |
| Peer Membership Protocol | SOAP - XML Message structure |
| Peer Binding Protocol | UDDI - Discovery protocol |
| Peer Endpoint Protocol | WSDL - Service description protocol |

(그림 4) P2P 공용 서비스와 XML Web Services

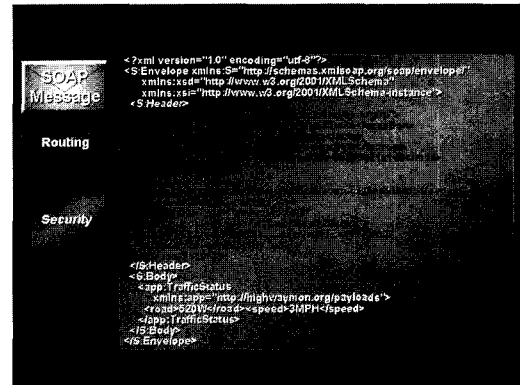
3.1.2 Global XML Architecture 설계 원칙

앞서 소개한 Global XML Web Services Architecture 하에서 정의되는 표준들은 다음과 같은 4가지 설계 원칙을 따른다.

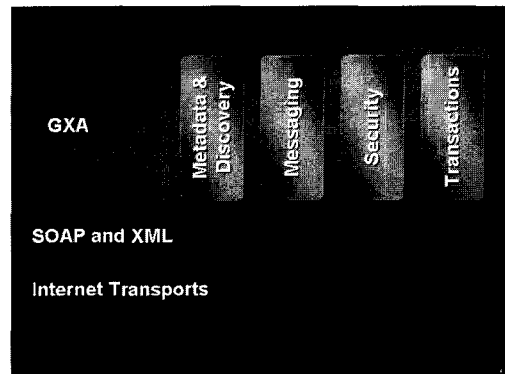
- **모듈화:** 완전한 기능을 가진 커다란 하나의 덩어리가 아니라, 각각의 기능을 가진 작은 단위의 컴포넌트들로 만들어져야 한다. 이것은 필요 없는 기능을 제공하거나, 꼭 필요한 기능을 누락시키지 않으면서 기능을 필요로 하는 곳에게 제공되는 하나의 솔루션으로 다시 묶여져야 한다. 이렇게 모듈화되어 있는 형식으로 아키텍처를 구성하는 것은 새롭거나 확장된 기능을 제공할 필요가 있을 때 신속하게 접근할 수 있는 장점을 가지게 된다. 새롭거나 확장된 기능들이 요구될 때 모듈러 프로토콜이 만들어질 수 있다.
- **범용성:** GXA 표준들은 B2B, EAI 솔루션, P2P 그리고 B2C에 이르는 광범위한 범위의 아키텍처에 사용되기 위해서 디자인 되었다. 각각의 모듈은 개별적으로 사용되거나, 아니면 다른 모듈과 조합되어 사용될 수 있으며 애플리케이션 도메인에 대해 비종속적이어야 한다.
- **연합성, 결합성:** GXA 표준들은 중앙서버나 중앙화된 관리기능을 요구하지 않아야 한다. 또한 실제 메시지를 수신하는 영역에서 사용하

는 기술에 대해서 아무런 가정을 하지 않는다. 어떤 기술이라도 사용될 수 있어야 한다.

- **표준에 기반한 설계:** GXA는 XML 웹 서비스의 기본적인 표준인 SOAP, WSDL, UDDI 기반이어야 한다. 또한 Microsoft사는 XML Web Services의 상호운용에 필요한 미래의 표준을 표준화하기 위해서 업계의 파트너들과 함께 작업할 준비하며, 표준 기구를 통해 발표한다.



(그림 5) GXA 표준을 사용한 SOAP 메시지 예



(그림 6) 주요 GXA 표준

3.1.3 GXA의 기능(현재)

GXA는 웹 서비스 기반의 애플리케이션을 구축할 수 있는 프로토콜 플랫폼 역할을 하며 다음과 같은 플랫폼의 핵심 기능을 제공한다.

애플리케이션이 인식하는 가상 네트워크: GXA

의 중요한 가치는, 이것이 임의적인 네트워크 전송과 프로토콜 상에서 가상 네트워크를 제공한다는 점이다. “웹 서비스 전화”를 제공하는 GXA 프로토콜이 WS-Routing이다. WS-Routing은 메시지들이 최종 수신자에게 전달되기 전에 한 개 이상의 라우팅 노드를 거쳐갈 수 있도록 하여, SOAP/1.2에서 정의된 “SOAP 매개”의 개념을 활용한다. WS-Routing은 임의의 전송장치 상에서 동작하기 때문에, 노드에 전달될 때마다 다른 전송장치를 사용하게 될 수도 있다.

WS-Routing을 사용하는 메시지들은 여러 개의 hop-to-hop 전송장치를 통해 전송될 수 있기 때문에, WS-Routing 자체로서는 매개체가 존재하는 경우 데이터그램 수준의 기능성만을 보장할 뿐이다. 어떠한 경우에도 안정적인 전송을 가능하게 하기 위하여, 마이크로소프트는 현재 임의적인 전송장치를 통해 메시지를 안정적으로 전송할 수 있는 메시징 프로토콜을 정의하고 있다.

GXA의 네트워크 가상화(virtualization)를 살펴보면 “TCP/IP를 사용하면 되는 것 아닌가”라는 질문을 하게 될 것이다. 첫째 이유는, 애플리케이션 고유의 최적화를 가장 상업적인 TCP/IP 스택에 구현하는 것은 너무나도 어렵다는 것이다. 네트워크 프로토콜을 XML로 표현하게 되면, 기존의 XML 기반의 틀과 기술을 활용하는 애플리케이션 고유의 하부구조(라우터, 게이트웨이, 브릿지 등)를 더 쉽게 구현할 수 있게 된다. 이 밖에, 새로운 라우팅 알고리즘 또는 재전송 휴리스틱스(retransmission heuristics)를 구현하는 것은 사용하기 쉬운 틀을 통해 사용자 모드의 코드로 구현되기 때문에, 프로토콜 호출이 더 이상 저수준 커널의 고난이도 작업이 아니다.

메시지 지향의 보안: SOAP이 매개 개념을 정규화시켰기 때문에, 전송장치 수준의 보안을 이용하는 능력은 더 이상 합리적인 것이 아니다. 매개체는 메시지를 다음 매개체 또는 수신자에게 전달하

기 전에 메시지를 처리해야 하기 때문에, 웹 서비스 보안은 메시지가 SOAP를 지원하는 정도를 명시적으로 알아야 한다. GXA의 기본 보안 프로토콜인 WS-Security를 통해 이것이 가능하다.

WS-Security는 SOAP 메시지의 서명(signing)과 봉합(sealing) 부분을 위한 SOAP 인식 메커니즘을 제공한다. 특히, WS-Security는 발신자의 보안 상장을 전달하는 SOAP 헤더를 정의한다. 이 상장은 발신자의 신원을 확인하는 한 개 이상의 자격을 대표하는 것이다. 메시지 수신자는 이 자격을 사용하여 메시지가 처리될 수 있는지를 판단한다. 발신자가 필요한 자격을 충족시키지 못한다면, 메시지는 잘 알려진 오류 코드와 함께 거부될 것이다. 반대의 경우라면, 메시지는 기대하는대로 처리될 것이다.

WS-Security는 보안 상장에 대해 특정 포맷을 강요하지는 않고, 보안 헤더의 확장가능 요소를 통해 특정 포맷이 지원된다. WS-Security는 광범위하게 사용되는 여러 프로토콜(X509, Kerberos 등)에 대한 요소 포맷을 미리 정의하고 있다. WS-Security 헤더에 있는 보안 상장은 서명과 봉합을 위해 사용된다. WS-Security는 SOAP 메시지의 서명과 봉합 부분을 지원한다. 메시지를 서명하게 되면, 발신자가 보안 상장을 처리한 사람이라는 것과 메시지의 서명된 부분이 조작되지 않았다는 것을 보장하게 된다. 메시지를 봉합하는 것은 발신자와 수신자만이 이해하는 키를 사용하여 메시지의 내용을 암호화하는 것이다. WS-Security는 서명하고 봉합할 수 있는 프레임워크를 제공하고 실제 알고리즘을 위하여 XML Signature, XML Encryption과 같이 외부에서 정의된 프로토콜을 참조한다. 그렇기 때문에 WS-Security는 새로운 알고리즘이 개발될 경우에 이를 적극적으로 수용할 수 있다.

3.1.4 GXA의 미래

- 연합 보안(federated security)에 대한 전폭적인 지원: WS-Security는 안전한 메시지를 보내는

데 필요한 최소한의 기능을 대표하는 기본 명세다. WS-T rust는 보안 상장을 발행하고 신뢰 관계를 관리할 수 있는 프로토콜을 정의할 것이다. WS-Privacy는 정보보호의 요구사항과 선호사항에 대해 명시할 수 있는 프레임워크를 제공할 것이다. WS-Authorization은 서비스의 자격/권한 요구사항을 발견할 수 있는 메커니즘을 제공할 것이다. WS-Federation은 무기명 조직과 인증 기술 간의 신뢰 관계를 중재하는데 필요한 프로토콜을 정의할 것이다. WS-SecureConversation은 메시지 내용을 봉합하는데 적합한 안전한 세션 키를 만드는 것과 같이, 안전한 통신을 구성할 수 있는 프로토콜을 제공할 것이다. 마지막으로, WS-Policy는 범용적인 방식으로 서비스의 기능과 요구사항을 발견할 수 있는 프레임워크를 제공할 것이다.

- 분산 계약: SOAP 초기부터, 사용자들은 웹 서비스 환경에서 어떤 역할 트랜잭션이 가능할 것인지 궁금해왔다. 광범위한 애플리케이션의 요구사항을 충족시키기 위하여, 마이크로소프트는 GXA 기반의 분산 계약 프로토콜을 만들 수 있는 프레임워크를 정의하고 있다. 이 프레임워크는 특정 계약 프로토콜을 강요하지 않고 여러 당사자들이 협업을 할 수 있는 공통 프레임워크만을 정의한다.
- 영속적인 메타데이터와 발견: GXA는 메타데이터가 어디에나 산재해있고 신뢰성이 높다는 것을 가정하고 있다. 이 가정을 위하여, 마이크로소프트는 웹 서비스 애플리케이션을 구성하는 다양한 실체들에 대해 임의적인 설명을 할 수 있는 XML 기반의 메타데이터 프레임워크를 개발하고 있다. 이 프레임워크는 종착점과 메시지에 대한 기능, 선호사항, 요구사항을 표현하는데 사용된다. WS-Referral은 이 프레임워크의 첫 번째 애플리케이션이며, 향후에 나올 명세는 메타데이터의 역할을 다른 영역으

로 확대할 것이다.

3.2 .NET 2기의 새로운 프로젝트들

마이크로소프트의 .NET은 어느 한 시점을 목표로 완성하는 것이 아니라 계속 진화하고 수정되는 장기간의 비전이다. 다음과 같은 새로운 제품과 기술이 GXA 등을 기반으로 개발되고 있거나 곧 출시될 예정이다.

- Greenwich

Greenwich는 실시간 커뮤니케이션 및 협업 서버(real-time communications and collaboration (RTC) server)로 사용자들이 디바이스에 관계없이 고급 멀티미디어 컨퍼런싱 기능을 사용할 수 있도록 지원한다.

- Yukon

새로운 SQL 서버의 코드명으로 XML 지원과 분산 데이터(distributed data) 관리 기능을 강화하였다.

- Everette

새로운 Visual Studio .NET의 코드명

- Longhorn

새로운 윈도우 OS의 코드명으로 2004년 발표 예정이다.

- Windows Media Center

PC 기반 TV로 닷넷의 개념이 컴퓨팅을 뛰어 넘어 다양한 디지털 엔터테인먼트와 디지털 콘텐츠에 적용하는데 핵심 역할을 할 것으로 기대된다.

- Web Matrix

신개념 웹개발툴로 현재 마이크로소프트 웹사이트에서 베타 버전을 무료로 다운로드 할 수 있다.

- Server Manager Project

Microsoft Operations Manager와 Application Center에 기반한 프로젝트로 웹기반 어플리케이션의 엔드-투-엔드(End-to-End) 관리를 가능케 할 것이다.

- MSN 8

올 가을 선보이게 될 새로운 MSN으로 PassPort,

.NET Alerts 등의 기능과 함께 오프라인 기능을 강화하였다. 특히 스팸 방지, 온라인 보안, 바이러스 예방과 같은 보안 관련 기능을 강화하였다.

4. 결 론

10년 전, 마이크로소프트는 “손끝에 모든 정보를”이라는 세계적인 비전을 발표했었다. 그 때를 상기하면, 정보는 4800bps로 연결된 모뎀에 불과하였으며, 대부분의 메시지는 이메일이 아닌 팩스로 전송되었고 인터넷을 사용하는 사람은 거의 없었다. 마이크로소프트는 사람들이 원하는 정보를 원하는 시간에, 원하는 장비를 통해 이용할 수 있는 세계를 제시했지만, 기업들이 어떤 표준을 적용할 것인지에 대해서는 알지 못했다. 이제, 마이크로소프트는 XML이라는 표준을 제시하고 있다.

Microsoft .NET 플랫폼은 개방형 표준을 활용하는 첫 번째 플랫폼으로서 21세기의 컴퓨팅과 통신을 혁신적으로 바꿀 것이다. Microsoft .NET은 이전과 비교할 수 없을 정도로 컴퓨팅과 통신을 단순하고도 사용하기 쉽게 만들 것이다. 이 비전은 차세대 인터넷 서비스를 탄생시킬 것이며, 수 십만 명의 소프트웨어 개발자들이 혁신적인 온라인 서비스와 비즈니스를 구현할 수 있게 할 것이다. 이 비전은 사용자의 생활에 사용되는 기술을 사용자가 통제하고, 프라이버시, 보안과 데이터를 자신이 원하는대로 통제할 수 있게 할 것이다. 그리고 스마트 장비에서 실행되는 스마트 소프트웨어는 이 모든 것을 가능하게 할 것이다.

.NET에 대한 반응은 매우 다양하다. 유닉스의 뿌리가 깊은 국내에서는 .NET을 클라이언트만을 위한 가벼운 기술로 보며 Sun 또는 Java의 기술에 치중하는 시각, XML Web Services까지 마이크로소프트의 단일 플랫폼에 종속될 수 없다는 시각이 대표적으로 부정적인 의견들이다. 앞에서 충분히 설명했듯이, .NET은 클라이언트만을 위한 기술이 절대

로 아니다. 더욱이, XML Web Services의 표준 기술을 마이크로소프트가 IBM과 함께 개발하고 제정하고 있으며, Sun과 같은 전통적인 기술기업들이 이를 자신의 제품 또는 아키텍처에 반영하고 있는 실정이다. 적어도 마이크로소프트의 기술과 제품은 일반 사용자 뿐만 아니라 기업을 포함한 IT의 모든 부문에서 절반 이상에 영향을 미치게 될 것이다. 그렇기에 .NET은 절대로 자신과 무관한 기술이 아니다. 그렇다고 종속을 경계하는 시각도 지나친 감이 있다. XML Web Services는 기술과 제품에 상관 없이 서로 연동되는 통합에서 시작된 것이다. 그렇기 때문에, 새로운 플랫폼을 위하여 그리고 유지보수를 위하여 한 곳으로 IT 환경을 집중시켰던 과거 그리고 현재에 비해 매우 유연해지게 된다. .NET에서는 VB, VC++와 같은 언어뿐만 아니라 COBOL, Pascal을 비롯한 20개 개발언어를 사용할 수 있다는 것이 그 한 예다.

XML Web Services는 대한민국에게 매우 중요한 전환기가 될 것이며, .NET은 그 중 가장 앞서고 가능성이 높은 플랫폼이 될 것이다.

저자 약력



오 세 영

- 1994년 한국외국어대학교 경영정보대학원(경영학 석사)
- 1998년 마이크로소프트 비즈니스 솔루션 그룹 입사
- 2002년 마이크로소프트 닷넷 팀 차장