5

웹 2.0 응용 및 플랫폼 기술의 진화

(주)넥스알 ▌한 재 선

1. 서 론

"참여", "공유", "개방", "분산" 등을 핵심 철학으로 하는 웹 2.0은 정보의 생산과 유통에 많은 변화를 가져 왔다. 정보 생산의 중심축이 서비스 제공자에서 일반 사용자로 넘어가면서 정보의 양적 폭발을 일으켰고 질적인 측면에서도 프로추어(proteur)의 등장과 확산으로 양질의 정보를 생산해 내고 있다. 이렇게 생산된 정보는 다양한 채널을 통해 공개됨으로써 정보의 공유가 용이해 지고 이로써 정보가 재생산되는 정보 생산과 유통의 선순환 구조를 만들었다. 하지만 이것은 웹 기술의 꾸준한 진화로 인해 기술적 성숙도가 어느 정도 수준에 도달했기 때문에 가능한 것이다. 그림 1은 웹 2.0 핵심철학들과 그들이 기반으로 하고 있는 기술들에 대한 예를 보여주고 있다. 블로그와 위키는 사용자의 참여에 의한 정보 생산을 가능하게 하였고 PageRank, 추천 시스템 등은 사용자의 집단지성을 끌어내어 보다 가치있는 정보를 만들어 내는데 큰 역할을 했다. RSS와 Open API는 각각 데이터와 서비스를 공개하는 표준 포맷을 제공하여 다양한 Mashup 서비스가 시도되는 기반을 제공하고 있다. 추천 시스템과 개인화 기술들은 정보와 상품의 분산화 된 소비를 가속화시켜 롱테일 현상을 낳고 있다.

웹 2.0 기술은 어느 순간 갑자기 등장한 것이 아니라 끊임없이 진화하고 있는 웹 기술의 연장선 상에 있다는 것을 주목할 필요가 있다. 그렇다면 앞으로 웹 기술은 어떻게 진화할 것인가? 그에 대한 해답이 바로 "플랫폼으로서의 웹(Web as Platform)"이다. 즉, 단순 정보 전달의 역할을 하던 웹이 보다 다양한 서비스를 개발하고 운영할 수 있는 환경으로 진화한 것이다. 기존에 PC 응용프로그램을 개발하기 위해 윈도우즈 OS가 개발과 운영에 필요한 환경을 제공해 주었다면 이제는 웹이 동일한 기능을 서비스로 개발할 수 있는 환경을 제공해 주고 있다. 웹이 플랫폼의 역할을 할 수 있게 된 것은 클라이언트와 서버 기술, 웹 개발 기술, 데이터 처리 기술, 글로벌 플랫폼 기술 등 전

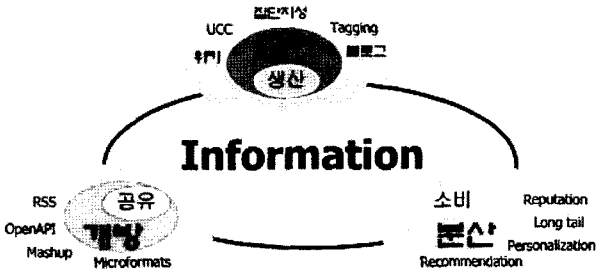

그림 1 정보 유통을 위한 웹 2.0 기술들

기술 스택에 걸쳐 기술적 진보가 있었기 때문이다. 본 논문에서는 웹 2.0이 참여, 공유, 개방의 성격을 가질 수 있게 한 기술들과 함께 웹이 플랫폼으로 역할을 할 수 있게 한 기술적 토대에 대해 살펴본다.

2. 참여의 기술

웹 2.0시대에는 사용자의 역할이 단순 컨텐츠 소비자에서 생산자로서의 역할까지 확대되었다. 컨텐츠 생산자 풀의 엄청난 확대로 웹 컨텐츠는 기하급수적으로 늘어났다. 더욱이 사용자에 의해 생산된 컨텐츠가 서비스 제공자 컨텐츠보다 더 가치를 가지는 역전현상까지 일어나고 있다. 이러한 사용자 생산 컨텐츠를 통칭하여 UCC(User-Created Contents) 혹은 UGC(User-Generated Contents)라고 한다. 블로그와 위키 등의 텍스트 데이터를 시작으로 사진, 음성, 동영상 등의 멀티미디어 데이터까지 UCC의 영역이 확대되고 있다. 동영상 UCC 서비스가 핵심으로 떠오르면서 코덱, 스트리밍 등의 동영상 기술이 크게 주목 받고 있다. 또한 대용량 스토리지와 효율적인 네트워크 기술이 동영상 UCC를 위한 인프라 기술로 활발히 연구되고 있다.

UCC가 개인적인 목적을 위한 사용자의 참여라면 어떤 하나의 공동의 목적을 위해 사용자의 참여를 활용하는 경우가 있다. 예를 들어 위키피디아[1]는 온라인 백과사전으로 누구나 참여하여 컨텐츠를 수정할 수 있고 모든 수정 버전을 유지할 수 있는(revision log) 위키 기술이 밑바탕이 되고 있다. 명시적인 사용자 참

여가 필요한 위키피디아와는 달리 사용자의 개인적인 행위들에서 가치있는 정보를 끌어내 공동의 목적을 달성하는 방법도 있다. Google의 PageRank[2], Amazon.com의 추천 시스템[3], Yahoo의 Tag Suggestion[4] 등이 이에 해당한다.

사용자의 참여가 직접적이든 간접적이든 개별적인 지식을 모아 공동의 목적을 달성할 수 있는 거대한 지식이 도출된다. 이것을 집단지성(Collective Intelligence)라고 한다. 집단지성은 개별적인 지식의 단순 합보다 더 크다고 한다. 이것은 참가자들 사이의 네트워크 효과로 인한 것으로 이를 어떻게 효과적으로 끌어낼 것인가가 집단지성의 핵심이다. James Surowiecki의 "The Wisdom of Crowds"[5]에서 집단지성이 발현될 수 있는 마지막 요구조건인 "Good method for aggregating opinions" 역시 같은 맥락에서 이해할 수 있다. 따라서 "참여"의 웹 2.0에서 핵심 경쟁력은 사용자들의 개인적인 행위에서 가치 있는 집단지성을 끌어낼 수 있는 시스템이라 할 수 있다. 지금부터 그런 시스템에 대한 예를 살펴본다.

2.1 Google의 PageRank

Google 이전 검색엔진의 문제점은 검색 결과의 순서가 문서의 가치를 제대로 반영하고 있지 못하다는 것이다. 이는 단순히 문서 내에서 검색 키워드의 빈도수에 기반하여 순서를 정함으로써 스팸 문서에 취약하기 때문이다. Google PageRank[2]는 기존 Ranking 알고리즘을 뒤엎은 것으로 기본 아이디어는 단순하다. 즉, 웹페이지 A가 웹페이지 B를 링크한 행위를 A가 B의 가치에 한 표 지지한 것으로 해석하여 유입되는 링크의 수가 많을수록 가치있는 페이지로 인식하는 것이다. PageRank 알고리즘은 집단지성의 대표적인 예로 언급된다. 사용자들이 각자의 페이지에 링크를 추가하는 것은 완벽히 개인적인 행위다. 하지만 이런 개인적인 행위에서 다른 페이지에 대한 평가라는 새로운 가치를 찾아내고 모음으로써 페이지에 순위를 정해주는 집단지성을 끌어낸 것이다.

2.2 Amazon.com의 추천 시스템

추천 시스템은 Amazon.com의 가장 중요한 성공요인 중 하나다. 추천 시스템을 통해 고객은 관심 분야의 새로운 책에 대해 알게 되고 묻혀 있던 명저를 발견하게 된다. 이것이 롱테일 현상[6]을 불러일으킨 하나의 기술적 장치다. 판매량 그래프의 꼬리에 놓여 있던 책들이 추천을 통해 소개됨으로써 매출 증가에 기여한다. 추천 시스템의 기반기술인 Collaborative Filtering [7]은 사용자 개개인의 상품 평가 데이터를 바탕으로 전체 사용자들의 유사도나 상품의 유사도를 계산해 낸다. 즉, 상품 평가라는 개인적인 행위에서 유사도, 더 나아가서는 추천 시스템이라는 집단지성을 끌어낸 것이다. 검색엔진이 정보 검색을 위한 가이드 역할을 한다면 추천 시스템은 정보 탐색을 위한 가이드 역할을 한다. 따라서 정보 시스템의 한 축으로 중요한 위치를 차지하고 있다. 이를 입증하듯 Netflix라는 미국의 온라인 DVD 대여 서비스에서는 자체 추천 시스템의 성능을 10% 향상시키는 사람에게 100만불의 상금을 걸었다[8].

2.3 Social Bookmarking

기존 북마킹이 브라우저에서 실행되고 로컬 PC에 저장되는 반면 소셜 북마킹은 웹페이지에서 실행되고 웹에 저장된다. 이렇게 저장된 북마크는 검색 가능하고 다른 사람과 공유할 수 있는 장점을 가진다. del.icio.us[9]가 가장 대표적인 소셜 북마킹 서비스다. 소셜 북마킹 서비스가 지금과 같이 인기를 끌 수 있었던 것은 바로 태깅(Tagging)이라는 기술을 사용했기 때문이다. 즉, 북마킹할 페이지에 대해 사용자가 몇 개의 키워드를 붙이는 것이다. 이렇게 붙여진 태그는 검색과 탐색, 양쪽에서 사용될 수 있다. 특히 기계에 의해 자동으로 키워드가 추출되는 것이 아니라 사용자에 의해 직접 붙여지므로 다양한 문맥 정보가 붙을 수 있다. 태깅이라는 행위는 사용자가 자신의 북마킹 정보를 정리하기 위해서 하는 지극히 개인적인 행위이지만 이를 통해 각 페이지는 메타데이터를 얻게 되고 검색과 탐색이 가능해 진다. 이렇게 구축된 태그 검색은 문서 분석을 통해 인덱싱 하는 전통적인 검색엔진과는 다른 특징을 가지며 차별화된 검색 기능을 제공할 수 있다. 또한 태그들과 페이지 사이의 연관관계를 분석하여 클러스터링[10], 온톨로지[11] 등의 추가적인 기능을 구현하는 연구들도 있다. 태그들을 분석하여 PageRank 알고리즘과 유사한 아이디어로 태그 제안을 구현한 경우도 있다[4].

3. 개방과 공유의 기술

포털을 중심으로 전통적인 웹싸이트들은 자신들의 데이터를 폐쇄적으로 가두어 두고 외부 서비스에서 접근하기 어렵게 했다. 방문자수와 체류시간이 웹싸이트의 성능 평가에 대한 절대적인 기준으로 작용하던 상황에서 폐쇄적인 서비스는 필연적인 결과라 할 수 있다. 하지만 웹 2.0 시대의 새로운 룰은 데이터와 서비스를 개방하여 데이터 접점을 늘리고 서비스에 대한 영향력을 확대하여 궁극적으로 트래픽 증가와 매출의

증가로 연결시키는 것이다. 데이터와 서비스의 개방은 데이터와 서비스를 표준 포맷으로 기술할 수 있는 RSS와 Open API 기술 덕분에 가능하게 되었다.

3.1 RSS: 데이터 개방을 위한 기술

RSS는 Really Simple Syndication의 약자로서 업데이트된 최신 웹 데이터를 공개하기 위한 표준적인 XML 데이터 포맷이다. RSS를 통해 공개되는 각 데이터를 RSS 피드라 하고 이를 읽을 수 있는 프로그램을 RSS 리더라 한다. 폐쇄적으로 데이터를 가두어 두던 전통적인 웹싸이트에서 최신 내용을 확인하기 위해서는 직접 방문하는 방법뿐이었다. 하지만 RSS를 통해 최신 정보를 서비스하는 웹싸이트의 경우 RSS 주소를 복사하여 자신의 RSS 리더에 등록하면 마치 신문 구독(Syndication)하듯이 최신 정보를 받아 볼 수 있다. 블로그와 뉴스 싸이트를 시작으로 대부분의 웹싸이트가 RSS를 필수적인 기능으로 인식하는 추세며 이로 인해 웹싸이트의 직접 방문자수에만 의존하는 기존 웹 비즈니스 룰이 바뀌고 있다.

RSS는 웹 데이터를 표현할 수 있는 표준 포맷을 제공했다는 점에서 중요한 의미를 가진다. 표준 데이터 포맷으로 공개된 RSS 데이터는 사용자뿐만 아니라 다른 서비스에서 쉽게 접근할 수 있다. 이것은 데이터의 재사용성을 극대화하여 데이터 접점이 늘어나는 효과를 가져 온다. 개인화 홈페이지나 블로그 등에서 다른 웹싸이트의 RSS 피드를 직접 볼 수 있는 것이 그 좋은 예다. 또한 데이터의 구조가 정의됨으로써 데이터를 remixing할 수 있게 되어 이를 이용한 다양한 서비스들이 등장하고 있다.

초기 RSS는 블로그 포스트나 뉴스 기사를 주 대상으로 하였으나 점점 RSS를 통해 전송되는 정보의 형태가 다양해지고 있다. 예를 들어 검색 결과, 버그 리포트, 위키 업데이트 등이 있다. 또한 데이터 형태에 있어서도 텍스트 위주에서 점점 음성과 동영상과 같은 멀티미디어를 담는 방향으로 진화하고 있다. 음성 데이터를 실어 나르는 경우를 podcast라 하고 동영상 데이터의 경우 video podcast 혹은 vodcast라 한다. 이를 이용하여 DTV, 셋탑박스 등과 같은 홈 디바이스와 휴대폰 등과 같은 모바일 단말기에서도 RSS를 통해 컨텐츠를 구독할 수 있다. 즉, 모든 디바이스에 공통된 데이터 전송 방법을 제공한다.

3.2 Open API: 서비스 개방을 위한 기술

RSS가 데이터 개방을 위한 표준 포맷을 제공한다면 Open API는 서비스 개방을 위한 표준 포맷을 제공한다. 선구적으로 Open API를 비즈니스에 도입하여 성공한 대표적인 예가 Amazon.com이다. 방대한 양의 상품 정보를 Open API를 통해 외부 싸이트나 프로그램에서 검색할 수 있게 함으로써 Amazon.com의 상품이 노출될 가능성을 높이고 이는 매출 증대로 이어졌다. 글로벌 포털들과 신생 웹 2.0 서비스들 역시 대부분 Open API를 통해 서비스를 공개하고 있다. Google은 34가지 서비스에 대해 API를 제공하고 있고 eBay와 Amazon.com은 API 공개를 통해 실제 수익을 얻고 있다. 국내에서도 주요 포털을 중심으로 공개되고 있는 서비스가 늘고 있는 추세에 있다.

Open API는 SOAP(Simple Object Access Protocol), REST(Representational State Transfer), Javascript, XML-RPC, JSON-RPC 등의 다양한 기술로 구현 가능하다. 이 중에서 SOAP과 REST가 대세를 이루고 있고 최근에는 Google Maps와 같이 Javascript를 이용한 방법들이 늘어나고 있다. SOAP은 구조적인 완벽함에도 불구하고 복잡함으로 인해 Open API를 이용하는 개발자들은 보다 용이한 REST를 주로 이용하고 있다. 하지만 향후 기업 컴퓨팅 환경에 SOA(Service-Oriented Architecture)가 핵심으로 떠오르면서 웹 2.0과의 통합과정에서 SOAP에 무게가 실릴 가능성이 있다.

3.3 Mashup

데이터와 서비스가 개방되고 공유됨으로써 한 가지 이상의 웹 서비스가 합쳐져서 새로운 서비스를 만드는 Mashup 서비스들이 등장하고 있다. 최초의 매쉬업 서비스로 일컬어지는 HousingMaps.com은 Google Maps의 지도 서비스와 Craigslist의 부동산 정보 서비스를 "mashup"하여 부동산 정보를 지도 위에 보여줌으로써 새로운 가치를 가지는 서비스를 만들어 냈다. 이와 같이 매쉬업 서비스의 가치는 각 서비스의 강점을 살리고 부족한 점을 매쉬업을 통해 보강함으로써 개별 서비스가 가지지 못했던 새로운 가치를 제공한다는 것이다.

최근에는 Yahoo Pipes를 필두로 Google Mashup Editor, MS Popfly 등과 같이 매쉬업을 편리하게 할 수 있는 서비스들이 등장하고 있다. Yahoo Pipes는 코딩 한 줄 없이 기능 모듈을 잘 조합하기만 하면 새로운 매쉬업 서비스를 만들어 낼 수 있다. 또한 매쉬업된 서비스를 호스팅하고 서로 공유할 수 있게 한 것도 이들의 공통된 특징이다. 프로그래밍에 대한 지식이 전혀 없이도 매쉬업 서비스를 만들어 낼 수 있게 하여 매쉬업에 대한 저변을 확대하는 효과를 가져 온다. 데이터와 서비스를 RSS와 Open API를 통해 단순히 개방하는 소극적인 접근에서 한발 더 나아가 RSS와 Open

API를 활용하는 프로세스까지 서비스화 하여 제공하는 적극적인 방향으로 발전하고 있다. 즉, 매쉬업 서비스가 플랫폼화 되어가고 있다고 해석할 수 있다.

매쉬업 서비스의 플랫폼화는 지리 정보 서비스 영역에서 두드러지게 나타나고 있다. MS Research의 SenseWeb 프로젝트[12]는 다양한 센서 데이터를 MS Virtual Earth 위에 맵핑하여 실시간으로 볼 수 있게 하는 일종의 매쉬업 서비스다. 여기서 주목할 점은 데이터 소스인 센서는 MS에서 직접 설치한 것이 아니고 다양한 기관에서 연구나 공공용으로 설치한 것이라는 점이다. SenseWeb은 그런 데이터를 지도 서비스 위에 공개할 수 있는 공통 플랫폼을 제공하는 것이다. 또 다른 예로 GeoCommons 서비스는 사용자들이 업로드한 지리 정보를 데이터베이스화하여 다른 사용자들이 검색하여 지도 서비스와 매쉬업할 수 있는 플랫폼을 제공한다. 지리 정보를 한 곳에 모아 공유할 수 있고 지도 위에 다양한 지리 정보를 동시에 맵핑해 볼 수 있는 장점을 가지고 있다. 소프트웨어에서 오픈 소스의 역할처럼 데이터에서 오픈 데이터의 역할을 한다고 볼 수 있다.

4. Web as Platform

전통적인 웹 서비스 개발 방법론에서는 새로운 서비스 개발을 위해 데이터베이스를 구축하고 여기에 직접 컨텐츠를 생산해 쌓는다. 하지만 웹이 플랫폼화 되면서 웹에는 엄청난 양의 컨텐츠가 쌓이고 있다. 물론 많은 부분이 사용자 참여에 의해 생산된 것이고 앞으로 점점 더 늘어날 것이다. 또한 웹 컨텐츠는 RSS와 Open API를 통해 외부로 표준화된 인터페이스를 제공한다. 이것은 마치 데이터베이스를 이용하기 위해 SQL이라는 표준 문법을 사용하는 것과 흡사하다. 이와 같이 웹은 이미 그 자체로 데이터베이스의 역할을 충분히 할 수 있는 환경이 되었으며 따라서 그 위에서 개발자, 제휴 싸이트, 외부 서비스 등에 의해 새로운 프로그램이나 서비스가 만들어질 수 있다.

이러한 웹 플랫폼 환경을 제공해 주는 역할을 하는 것이 플랫폼 기업이다. 그 대표적인 예가 eBay와 Amazon.com이다. eBay는 경매와 오픈마켓을 기본으로 전자결제를 위한 PayPal, 세계 최대의 P2P VoIP Skype, 온라인 가격비교 서비스 Shopping.com, 쇼핑몰 제작 서비스 ProStores 등을 구비하여 온라인 상거래 서비스를 만드는데 필요한 토털 솔루션을 제공하고 있다. 실제로 3만명의 등록된 외부 개발자들이 eBay 상품을 검색하고 구매하는 프로그램과 서비스를 개발하고 있으며 45%에 이르는 eBay 상품들이 API를 통해 파트너 싸이트에 노출되고 있다. Amazon.com은 한 발 더 나아가 인프라 서비스까지 Open API를 통해 저렴한 비용으로 이용 가능하게 하고 있다. Amazon S3는 저렴한 비용에 안정적이고 확장성있는 스토리지 서비스를 제공하고 Amazon EC2는 컴퓨팅 자원을 공급하고 이용한 만큼 비용을 지불하게 하는 그리드 서비스를 제공한다. 이와 같이 개방형 인프라 서비스까지 갖춤으로써 데이터뿐 아니라 서버 자원까지 직접 구축할 필요없이 조합하여 새로운 서비스를 만들 수 있는 환경이 되었다. 이와 같이 웹이 플랫폼으로 진화할 수 있게 된 것은 다양한 측면에서 웹 기술이 발

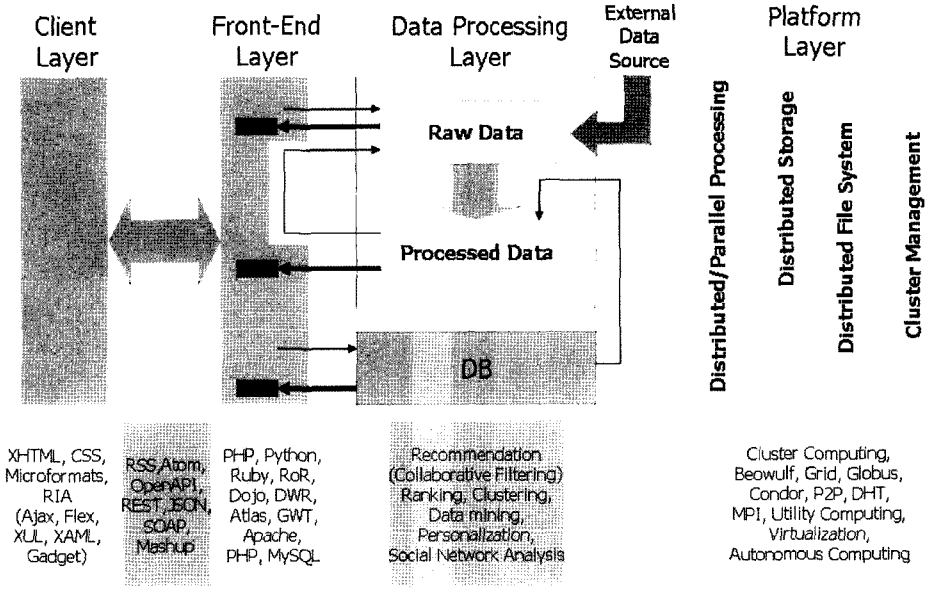

그림 2 웹 플랫폼을 구성하는 기술 구조

전했기 때문이다. 그림 2는 웹 플랫폼을 구성하는 기술 스택의 구조를 나타낸다.

4.1 Rich Client 기술

웹에서 클라이언트는 기본적으로 브라우저를 말한다. XHTML, CSS, DOM, Javascript 등의 웹 표준 기술의 발전으로 웹페이지의 표현력과 동적 구성능력이 향상되었다. 그러나 브라우저 기반의 웹페이지는 PC 어플리케이션과 비교했을 때 사용자 인터페이스와 Interactivity 측면에서 여전히 부족한 것이 사실이다. 이러한 제약사항을 극복하기 위한 기술이 RIA(Rich Internet Application) 기술이다.

웹 2.0과 함께 가장 급부상한 RIA 기술로 Ajax(Asynchronous JavaScript and XML)[13]가 있다. Ajax는 웹에서 비동기 통신이 가능하게 하여 웹페이지의 동적인 변화를 가능하게 하였다. Ajax를 구성하고 있는 CSS, DOM, XML, XSLT, XMLHttpRequest, Javascript 등의 기술 스펙에서도 알 수 있듯이 Ajax는 전혀 새로운 기술이 아니라 기존 표준 웹 기술을 묶어서 새롭게 정의한 것에 불과하다. 하지만 Google이 Ajax 기술을 이용하여 Google Maps와 Gmail 등의 어플리케이션을 웹 서비스로 성공적으로 구현함으로써 Ajax의 우수성이 입증되었고 웹 2.0을 표방하는 많은 서비스들이 Ajax를 도입하고 있는 추세다.

하지만 Ajax 이외에도 다양한 RIA 기술들이 등장하여 리치 클라이언트 플랫폼을 장악하기 위해 경주하고 있다. 여기에는 Flash, Flex, Apollo, Silverlight, OpenLaszlo, JavaFX 등이 있고 이들은 대부분 운영체제와 브라우저에 종속되지 않는 기술을 제공하며 일부는 오픈소스 전략을 취하고 있다. 이들 RIA 기술의 경쟁 속에 웹 클라이언트의 한계는 점점 극복될 것이고 결국 PC 어플리케이션과 웹 어플리케이션 간의 구분이 사라지게 될 것이다.

4.2 Open Platform 기술

데이터와 서비스를 개방하기 위한 기술로서 앞서 다루었던 RSS와 Open API, Mashup 등이 여기에 해당한다.

4.3 Web Development 기술

웹 서비스의 프론트엔드에 해당하는 기술로서 웹 개발을 용이하게 해 주는 웹 프레임워크 기술과 기본적인 서비스 운영 환경이 여기에 속한다. RoR(Ruby on Rails), CakePHP, Django, Atlas, DWR 등의 웹 프레임워크는 대부분 MVC(Model-View-Control) 패턴의 웹 개발을 지원하고 Ajax와 동적 효과 등을 편리하게 사용할 수 있는 Javascript 라이브러리를 가지고 있다. 이를 이용하면 개발시간을 현격히 단축할 수 있을 뿐 아니라 향후 유지보수와 디버깅에도 여러 가지 장점을 얻을 수 있다. 최근 서비스 운영 환경으로는 오픈소스 소프트웨어들이 환영받고 있다. 웹 서버는 Apache, 데이터베이스는 MySQL, 서버 운영체제는 Linux, 개발 언어는 PHP, Python 등의 스크립트 언어를 사용하고 이들을 묶어서 LAMP라 한다.

4.4 Data Processing 기술

웹에 데이터가 점점 많아지고 복잡해지면서 가치있는 정보를 찾아내는 것이 중요해 지고 있다. 지금의 웹 2.0 서비스들은 쏟아져 나오는 데이터를 그저 안정적으로 담아주는 것에서 만족하는 수준이다. 하지만 서비스의 경쟁력은 데이터의 양보다 데이터의 질에서 찾아야 하고 이를 위해서 데이터 처리 과정이 필요하다. 앞서 논의한 집단지성을 이용한 검색 랭킹이나 추천 시스템, 소셜 북마킹 등이 데이터 처리 기술의 좋은 예이며 이외에도 소셜 네트워크 분석, 평판 시스템, 스팸 처리 기술, 멀티미디어 자동 태깅 등이 모두 원본 데이터에서 새로운 가치를 끌어내는 데이터 처리 기술로서 서비스의 핵심 경쟁력이 될 수 있다.

4.5 Global Service Platform 기술

Google의 성공 요인은 PageRank라는 우수한 랭킹 알고리즘과 함께 그것을 계산해 줄 수 있는 분산 컴퓨팅 환경 덕분이다. Google은 전통적으로 고가의 서버 대신 저가의 PC 서버를 클러스터링하여 사용한다. 그래서 운영 비용은 최소화 할 수 있는 반면 안정성에서 항상 문제점을 안고 있다. 이것을 Google은 똑똑한 시스템 소프트웨어를 개발하여 해결하고 있다. GFS(Google File System)[14]라는 분산 파일 시스템은 기본적으로 모든 파일에 대해 세 개의 복사본을 유지하여 안정성을 보장하고 flat architecture로서 파일 분산을 효율적으로 함으로써 저장공간에 대한 utilization을 극대화한다. BigTable[15]은 웹 데이터와 같은 대용량 데이터를 효율적으로 저장하고 특히 읽기 성능에 최적화된 데이터베이스다. MapReduce[16]는 대용량 데이터 처리 작업을 수많은 서버들에게 지능적으로 분산하여 병렬처리 함으로써 빠른 성능을 낼 수 있게 한다. Google의 검색엔진이나 Google Earth, Gmail 등과 같은 실제 서비스가 모두 이러한 통일된 환경에서 개발되고 운영된다.

웹 서비스가 점점 글로벌화, 대용량화 되어 가면서 기초 체력에 해당하는 하부 서버 플랫폼에 대한 경쟁력이 중요해 지고 있다. 특히 비교적 저비용으로 시스템을 구축해야 하는 웹 환경에서는 저렴한 서버 하드

웨어 위에 지능적인 소프트웨어를 개발하여 구축 및 운영비용을 최소화 할 필요가 있다. 또한 데이터 처리를 통해 가치있는 정보를 끌어내는 기능이 점점 중요한 경쟁력으로 작용하고 있는 현실에서 이를 제대로 지원하기 위한 기반 플랫폼이 절실히 필요한 상황이다.

5. 결 론

웹 2.0의 성공과 발전은 그 밑바탕에 기반이 되고 있는 웹 기술의 발전에 기인한 것이라는 사실을 주목해야 한다. 그리고 현재의 다양한 웹 기술에 대한 깊이 있는 이해가 필요하다. 하지만 이러한 기술에 대한 이해가 단지 브라우저 기술이나 웹 개발 기술 등과 같이 외부에 직접 드러나는 측면에만 머무른다면 Google, Amazon, eBay 등과 같은 성공적인 케이스는 나오기 어려울 것이다. 데이터 처리, 글로벌 서비스 플랫폼 등과 같은 하부의 시스템 기술을 포함하여 전체 기술 스택을 조망할 줄 아는 안목이 필요하다. 또한 기술 컨버전스 시대에 발맞추어 웹 2.0 기술 역시 모바일 및 유비쿼터스 기술과의 융합을 고려하여 웹의 다음 모습을 그려 보고 준비하는 것이 필요한 때다.

참고문헌

[1] Wikipedia, http://www.wikipedia.org/

[2] Sergey Brin and Lawrence Page, "The Anatomy of a Large-Scale Hypertextual Web Search Engine", Proceedings of WWW7, Brisbane, Australia, 1998

[3] Greg Linden, Brent Smith, and Jeremy York, "Amazon.com Recommendations: Item-to-Item Collaborative Filtering", IEEE Internet Computing, January-February 2003

[4] Zhichen Xu, Yun Fu, Jianchang Mao, and Difu Su, "Towards the Semantic Web: Collaborative Tag Suggestions", Collaborative Web Tagging Workshop, Edinburgh, May 2006

[5] James Surowiecki, "The Wisdom of Crowds", Anchor, 2005

[6] Chris Anderson, "The Long Tail", http://www.wired.com/wired/archive/12.10/tail.html, Wired 12.10, Oct 2004

[7] Collaborative Filtering, http://en.wikipedia.org/wiki/Collaborative_filtering

[8] Netflix Prize, http://www.netflixprize.com/

[9] del.icio.us, http://del.icio.us

[10] Grigory Begelman, Philipp Keller, Frank Smadja, "Automated Tag Clustering: Improving search and exploration in the tag space", Collaborative Web Tagging Workshop, Edinburgh, May 2006

[11] Patrick Schmitz, "Inducing Ontology from Flickr Tags", Collaborative Web Tagging Workshop, Edinburgh, May 2006

[12] Andre Santanche, Suman Nath, Jie Liu, Bodhi Priyantha, and Feng Zhao, "SenseWeb: Browsing the Physical World in Real Time", Demo Abstract, ACM/IEEE IPSN06, Nashville, TN, April 2006

[13] Jesse James Garrett, "Ajax: A New Approach to Web Applications", http://www.adaptivepath.com/publications/essays/archives/000385.php, adaptivepath, 2005

[14] Sanjay Ghemawat, Howard Gobioff, and Shun-Tak Leung, "The Google File System", SOSP 2003

[15] Fay Chang, Jeffrey Dean, Sanjay Ghemawat, Wilson C. Hsieh, Deborah A. Wallach, Mike Burrows, Tushar Chandra, Andrew Fikes, Robert E. Gruber, "Bigtable: A Distributed Storage System for Structured Data", OSDI 2006

[16] Jeffrey Dean and Sanjay Ghemawat, "MapReduce: Simplified Data Processing on Large Clusters", OSDI 2004

한 재 선

2005 한국과학기술원 전자전산학과 공학박사
2005~2006 KAIST 전산학과 박사후연구원
2005~현재 GoogleSIG 모임 결성 및 운영자
2006~현재 웹 2.0 허브싸이트(www.web2hub.com) 운영
2007~2007 KAIST 정보미디어 경영대학원 대우교수
2007~2007 ETRI 텔레매틱스/USN 연구단 초빙연구원
2007~현재 ㈜넥스알 대표이사
E-mail : jshan0000@gmail.com