

Web 2.0 과 관련 기술

한 명 목*

◆ 목 차 ◆

- | | |
|--------------|----------|
| 1. 서 론 | 4. 관련 기술 |
| 2. 역사적 배경 | 5. 결 론 |
| 3. 웹 2.0의 특징 | |

1. 서 론

Web 2.0으로 정의된 World Wide Web(WWW)이 상당한 변화 중에 있으며, 관련 내용은 현재 깊게 논의되고 있다. 현재 진행되는 방향은 웹의 네트워크화, 즉 구조화가 진행되는 것이다.

Web 2.0은 2005년부터 가장 관심 있고, 중요한 논쟁거리 중에 하나가 되었다. 이 논문에서 현재의 변화가 다르고 새롭지만 어떤 공통적인 것을 가지고 있다는 사실에 큰 공감대가 왜 성립되었고 Web 2.0에 관련된 중요한 의미들을 검토한다. 또한 Web 2.0을 위한 매우 중요하고 영향력을 끼치는 기술인 AJAX에 대해서 설명하고, Web 2.0의 중요성에 대해서 논한다. 즉, 어떻게 새로운 기술들은 성공하는가? Web에서 중요한 역할자들은 어떻게 반응하는가? 어느 범위까지 그들은 이미 연결되어져 있는가에 대해서 논한다. 또한 Web에서 어떻게 새로운 것들이 계속해서 변하고 그것들을 예측할 수 있는지에 대해서 기술한다.

본 논문은 2장에서 Web 2.0에 대한 역사적 배경에 대해서 서술하고, 3장과 4장에서 주요한 특징들과 관련된 기술에 대해서 서술한다.

2. 역사적 배경

* 경원대학교 소프트웨어학부 부교수

Web 2.0 이라는 용어는 비교적 최근에 생겨났는데, 미국의 Tim O'Reilly 라는 출판업자가 2004년 10월경에 최근에 일어나고 있는 웹의 변화와 경향을 제2세대 웹, 즉 Web 2.0으로 표현하면서 비롯됐다.

끝이어 그들은 첫 번째 Web 2.0 Conference를 개최했으며, 2005년 10월 제2회 Web 2.0 Conference를 통해 네트워크 비즈니스 관계자들을 중심으로 급격하게 퍼져 나갔다. 현재 Web 2.0은 논쟁에서 뜨거운 토픽이며, 전통적인 방송 미디어에서 중요하게 다뤄지고 있다. 거기서 WWW에서의 새로운 기술들, 응용분야, 개념들, 아이디어 그리고 구현들에 대해서 설명하지만, 이것들은 방송과 기자들에 따라 내용이 다를 수가 있다.

Web 2.0이 Web 1.0에서 변화하는 데는 많은 요인이 작용했지만, 이것들을 크게 나누어 보면 질적 변화와 양적 변화로 볼 수가 있다. 질적 변화에는 HTML이 XML로 웹의 구조진화가 있으며, 양적 변화에는 웹의 사용자, 접속서버 그리고 데이터양의 증가가 있다.

<Web 1.0 과 Web 2.0 비교>

Web 1.0	Web 2.0
Double Click	Goole AdSense
Ofoto	Flickr
Akamai	BitTorrent
mp3.com	Napster

Britannica Online	Wikipedia
personal websites	blogging
evite	upcoming.org and EVDB
domain name speculation	Search Engine Optimization (SEO)
page views	cost per click
screen scraping	web services
publishing	participation
Content Management Systems(CMS)	wikis
directories (taxonomy)	tagging ('folksonomy')
stickiness	syndication

위의 표는 Web 2.0과 전통적인 Web 1.0과 기능별 차이를 비교해서 정리 한 것이다.

왼쪽에 있는 용어들은 부정적 이미지와 함께 일반적으로 성공하지 못하고, 시대에 뒤 떨어진 것으로 간주되어진다.

용어의 애매성을 반영하기 위해서 자세히 들어가기 전에 이 용어와 관련된 용어들의 모임을 한눈으로 그림 1과 같이 볼 수가 있다.

GTD

Web Standards Podcast Ajax Wiki

Folksonomies Blogging Web Service The Long Tail

Tagging Life-hacking Web2.0 RSS API Mash-up

Rich User Experience API's citizen media Collective intelligence

(그림 1) 관련 용어 모임

3. 웹 2.0 의 특징

3.1 풍부한 사용자 경험(Rich User Experience)

이 용어는 웹 어플리케이션의 행동에 관련된 사항이다. 새로운 웹 2.0 어플리케이션의 많은 경우 데스크 탑 어플리케이션과 매우 유사하기 때문에 Rich User Experience를 허용한다고 알려졌다. 후자는 사용자 소통에 기반 한 상태를 통해서 특징지어진다. 이것은 어플리케이션이 안 좋은 입력에 다른 상태를 가질

수 있고, 진행을 보여줄 수 있으며, 빨리 응답할 수 있다는 의미이다. 대신에 웹 페이지는 HTTP 프로토콜에 기반을 하는데, 이것은 상태들을 지원하지 않는다. 웹 페이지에 대해 각 사용자 행동 다음에 변화, 진행을 보여주기 위해 새로운 페이지가 생성되어야 한다는 것을 의미한다. 많은 웹 2.0 어플리케이션들은 AJAX 나 LAMP 등 오픈 소스 기술을 다양하게 사용하고 매쉬 업 함으로써 재활용하기 좋은 형식으로 서비스를 제공하고 있다.

3.2 사용자 참여(User Participation)

웹은 제공자와 사용자의 역할이라는 관점에서 바라보면 계속 변화하고 있다. 초기에 그것은 대부분 한 방향 미디어였다. 양 방향 사용의 기초는 사용자 참여가 피드백 형태로 가능해지면서 시작되었다. 또한, 메시징 보드와 게스트 북들은 참여하고 토론하기 위해 웹 사이트의 방문자를 허용한다.

웹에서 사회적 네트워킹의 현상은 일반적으로 한명의 특정한 호스트와 공통의 관심사항을 갖는 사용자들을 기반으로 하는 가상 커뮤니티들의 인기가 증가함과 동시에 시작되었다. 여기서, 사용자들의 프로필링, 메시징, 채팅, 토론들은 웹에서 급격하게 구축되었다.

Wikis의 아이디어처럼, 사용자 참여는 새로운 차원으로 시작되었다. 즉, 사용자들은 제한된 공간에서 위와 같은 것을 허용되어질 뿐 만 아니라 전체 사이트들의 콘텐츠를 창조하고, 수정하고, 삭제할 수가 있다. 오용은 콘텐츠의 전 버전의 도움으로 커뮤니티를 통한 빠른 정정에 의해 방지된다고 생각되어진다.

3.3 동적 콘텐츠(Dynamic Content)

초기의 사이트들은 정적으로 제공되었다. 이러한 예로는 개인 홈페이지와 관련된 웹 사이트들이 있다. 동적인 콘텐츠는 백업용 웹 사이트들의 데이터베이스 개발이 증가하기 시작하면서 소개되었다. 데이터베이스들은 CGI, Perl, PHP등과 같은 언어들로 쓰여진 HTML 확장 스크립트에 의해 접근되어졌다. 이런 한

응용의 예로는 온라인 가게, 가상 커뮤니티 플랫폼, 그리고 탐색엔진 등이 있다.

개인적인 홈페이지들이 점점 확장되고 웹 로그들이 의해 교체되어지면서, 동적 콘텐츠는 디자인 이슈가 되었다.

- 모든 엔트리는 URI를 통해 유일하게 식별되어진다.
- 오래된 콘텐츠는 삭제되어지지는 않고, 아키브와 permalink를 통해 접근이 가능하다.

이것은 또한 '모든 문장은 유일하고 안전하게 식별되어 진다'라는 것을 요구하는 오래되고 다른 하이퍼 텍스트 시스템 Xanadu의 주요한 이슈이다. URL과 같은 식별자들은 결코 무효화 되지 않는다는 것을 의미한다. 웹로그들은 또한 다음과 같은 특징을 구현한다.

3.4 메타데이터(Metadata)

웹 로그들에서 문서들은 그것들에 대한 하나 이상의 키워드가 주어진다라는 의미인 'tagged'이나 분류에 들어가는 것은 보통 있는 일이다. 이것은 의미적 가치를 높이는, 즉 기계가 읽을 수 있는 의미적 데이터의 창조의 아이디어에 속하며 시맨틱 웹의 중요한 이슈이다.

3.5 웹 표준(Web Standard -Valid Markup)

웹 표준들은 계속해서 W3C인 World Wide Web Consortium에 의해 개발되어졌다. 최종적으로 배포될 때, 그것들은 소위 Recommendation 로서 표현되어진다. W3C Recommendation에서 표현되어지는 문법을 따르는 웹 페이지들은 valid라고 불리며 W3C에서 유효성을 위해 검사되어질 수 있다. 웹 페이지의 유효성은 웹 2.0과 결합해서 주요한 특징으로 언급되어진다. 그러나 웹 2.0에서 실질적으로 어떻게 보여지는가? 한 웹로그 제작자는 웹 2.0사이트로 자주 언급되어지는 34개를 유효성을 위해 검사하였는데, 오직 2개만 에러 없이 통과하였다. 그래서 웹 2.0 은 아직 새로운 웹 표준 의식을 일으키지 않는 것 같다. AJAX 기술은 아직 W3C recommendation에 있는 XML 마크

업 언어와 DOM(Document Object Modelling) 트리의 공개된 포맷을 이용하며, 유사한 기능을 제공하는 flash의 닫힌 포맷과 특징적으로 다르다.

3.6 롱 테일(Scalability, "The Long Tail")

'The Long Tail' 이라는 용어는 peer-to-peer 기반의 어플리케이션들, 아마존, 이베이, 그리고 구글의 AdSense와 같은 웹 성공 스토리에 기여하는 한 특징이라고 할 수 있다. 그것은 어플리케이션을 쉽게 평가하고, 좀 더 많은 사용자들에게 더 큰 가치를 부여하는 능력을 포함한다.

아마존과 이베이의 성공 스토리를 점검한다.

- 아마존 : 사용자 상호작용을 소개했다. 예를 들면 사용자가 작성한 리뷰와 통계치를 활용한 사용자의 가능한 관심 제품의 그들의 유명한 추천 등이 있다. 이러한 특징들은 사용자들의 수를 가지고 쉽게 측정하며, 사용자들이 더 많으면 더욱 중요해진다.
- 이베이 : 대규모 제공과 수요의 원칙에 의존한다. 좀 더 많이 제공하면, 고객을 위한 서비스가 좀 더 흥미로워진다. 좀 더 좋은 고객이 있으면, 좀 더 매력적인 판매자를 위한 서비스가 있다.

이러한 예들은 인터넷을 위한 특별한 특징들인 'network effect' 잘 알려진 일반적인 성공의 공식을 보여준다. 이것은 경제 용어이며, 웹 2.0이나 웹을 이야기하기 훨씬 전에 언급되어져 왔다. 아마도 네트워크 효과를 위한 가장 오래된 예 중에 하나는 telephone network이다.

4. 관련기술

4.1 AJAX

4.1.1 개요

AJAX는 Asynchronous JavaScript and XML의 약어이며, 웹 페이지의 행동을 근본적으로 변화시키는 기술이다.

통상적으로 웹 페이지는 사용자 행동 후에 링크를 클릭하거나, 폼 형태에 데이터를 입력하거나 하면서 완전하게 다시 로드해야 한다. AJAX는 클라이언트 쪽 (웹 브라우저)에서 실행하는 엔진을 안내한다. 어떤 행동은 엔진 안에서 실행할 수 있으며 서버 쪽으로 데이터를 보낼 필요가 없다. 즉 그것들은 클라이언트 컴퓨터에서만 실행하고, 데스크 탑 어플리케이션들에 비해 매우 빠르다.

AJAX는 JavaScript를 포함한 다양한 기술들의 집합체이다. 종종, 엄청난 양의 JavaScript 코드가 Google의 이메일 서비스인 Gmail처럼 웹 페이지의 모든 사용이 가능하기 전에 로드된다. 그러면 웹 페이지는 Gmail인 경우 이메일을 통한 브라우저, 읽기, 이동, 답하기, 지우기 등과 같이 이미 많은 행동을 위한 필요한 대부분의 데이터를 포함한다. 오직 일부 데이터가 다시 로드하는데 필요하고, 이것은 SOAP과 같은 웹 서비스를 바탕으로 XML 사용을 가지고 비동기 Http 요청에 의해서 실행된다.

AJAX는 새로운 기술이 아니라 이전의 오래된 기술들의 집합이다. 수년 동안 웹 개발자들 사이에 중요하게 인식되지 않은 기술 중에 하나가 AJAX의 XMLHttpRequest를 제외한 유사한 기능을 제공하는 DHTML(Dynamic HTML)이다.

4.1.2 Code 예

JavaScript 코드는 페이지의 HTML <head> 안에 위치한다. 서버로부터 데이터를 받는 것은 현재의 버전에서 모든 주요한 브라우저에 의해 구현되는 XMLHttpRequest를 통해서 이루어진다.

먼저, JavaScript XMLHttpRequest object가 생성된다. 여기에는 인터넷 익스플로러와 모든 다른 브라우저를 위해 다른 두 가지 방법이 있다.

```
var requester = new XMLHttpRequest();
var requester = new ActiveXObject("Microsoft.XMLHTTP");
```

그리고 서버에의 비동기 연결을 열 수 있고 활동되어 진다.

```
requester.open("GET","/feed.xml");
requester.send(null);
```

데이터는 send() 메소드의 인수로서 보내진다.

이벤트 listener를 통해, XMLHttpRequest object의 상태가 점검 될 수 있고, 보내지는 것이 성공하면 구조화된 반응 성질들의 어떤 DOM에 데이터들을 포함할 수 있다.

사용자 입력 데이터는 확인 될 수 있다. 예를 들면 폼을 점검하기 위해 JavaScript를 이용한다. 이 방법은 사용자가 JavaScript의 onchange 이벤트를 일으키는 입력 필드 폼에서 멀어져 갈 때 불리어진다.

```
function checkForm() {
    if (!receipt.valid) {
        receipt.focus();
        alert("Please enter a valid receipt number.");
    }
    return false;
}
```

이것은 만약에 valid 변수가 false로 정해지면 JavaScript 팝업 경고 창을 연다. 이 변수를 정하는 것은 XML이 서버를 요청하는 방법으로 발생한다.

4.1.3 장점

웹 어플리케이션 행동은 데스크 탑 어플리케이션과 유사하다. 사용자 행동에 반응은 매우 빠르고 비록 HTTP가 상태 없는 프로토콜이지만 웹 어플리케이션은 상태를 가지고 있는 것을 닮았다. 웹에 다시 로딩 없이 콘텐츠가 바뀌든가, 요소들의 드래그와 드롭 이동 같은 이상한 행동들은 지금 가능하다. 이것은 전에 Macromedia 제품인 플래쉬 독점적인 형식으로만 가능했다.

그래서 브라우저 플러그인을 위한 필요 없이, AJAX는 어플리케이션기반의 플래쉬가 제공하는 것과 유사한 기능을 제공한다. 그러나 웹 사용자들의 65 퍼센트가 설치하는 브라우저 플러그인을 요구하는 동안

AJAX는 그러지 않는다. 그것은 웹 사용자의 10 퍼센트만 사용할 수 없는 JavaScript를 바탕으로 설치하며, 플래쉬 보다 상당히 작은 수이다.

4.1.4 단점

- JavaScript

앞에서 언급했듯이, AJAX 엔진은 JavaScript로 구성되었다. 따라서 현재 웹 사용자들의 10퍼센트는 AJAX 어플리케이션으로부터 소외되어있다.

JavaScript는 오직 마크업 언어인 HTML과 비교해서 완전한 프로그래밍 언어의 표현력을 가지고 있다. 이것은 AJAX에서 웹 어플리케이션이 프로그래밍 에러에 관련된 에러에 약하다는 것을 의미한다. 또한 요구되는 기능들이 작동하는지를 검증하기 위해 다양한 브라우저와 플랫폼에서 집중적인 테스트가 필요하다.

또한 JavaScript는 W3C에서 염려하는 이슈인 웹 접근을 위한 장벽을 구성한다.

JavaScript는 <noscript> tag를 가지고 항상 달라지게 추천된다. 그러나 이것이 실행되면 웹 페이지를 기반으로 하는 AJAX의 전체 기능을 전통적인 방법으로 제작성하든가, JavaScript를 사용하지 못하는 10퍼센트 정도의 사용자들한테는 가능하지 않다.

MS의 인터넷 Explorer에서 사용하는 XMLHttpRequest를 하는 사용자들 위해 AJAX 기능의 많은 부분인 활동적인 콘텐츠 모듈 ActiveX는 또한 4.1.2를 가능하게 해야 한다. 이것은 AJAX로부터 ActiveX는 인터넷 Explorer에서 알려진 보안 문제 때문에 많은 사용자들이 사용할 수 없게해서 많은 사용자들을 제외시킨다.

- The Back Button

웹 브라우저들은 주로 전체 웹 페이지의 기록을 가지고 있다. 사용자가 back button을 누를 때 사용자는 마지막 행동이 하지 않은 것을 기대한다. 이 기대는 하나의 행동이 브라우저를 위해 잡힐 수 있지 않기 때문에 AJAX 어플리케이션을 가지고 나타나는 영향을 확신하지 않는다. 현재 이 문제의 윤곽이 알려져 있고, 이러한 것은 AJAX에서 프로그램을 어렵게 한다.

- Bookmarking

많은 동적으로 생성된 페이지처럼, 어떤 한 페이지의 상태를 링크하거나 북마크하는 것은 거의 불가능하고, 이러한 상태들은 URL에 의해 유일하게 확인할 수 없다. 해결은 URL 부분 식별자의 사용처럼 더 많은 프로그래밍 노력이 결합된다.

4.1.5 AJAX 어플리케이션들의 예

- 구글의 새로운 서비스들

2004년과 2005년에 유명한 탐색 엔진 회사인 구글이 여러 개의 새로운 서비스를 소개했는데, 그 중에는 이메일 서비스인 Gmail, 지도서비스인 구글 Maps, 개인화된 탐색 페이지 등이 있다. 이 모든 것들이 AJAX 기술을 기반으로 한다.

- Window Live 데스크탑

MS사는 최근에 'Window Live'라는 데스크 탑용 어플리케이션인, 웹 기반의 베타 버전을 소개했다. 그것은 구글의 새로운 개인화 탐색 페이지를 닮았는데, 예를 들면 관리할 수 있는 미리 선택된 콘텐츠, 자유롭게 선택할 수 있는 콘텐츠, 웹 메일 등이 포함되어 있다.

- Flickr Photo 서비스

이 웹 사이트는 진행을 통보하고 그림 제목을 편집하도록 페이지에서 수정을 하기 위해 AJAX를 사용한다.

4.2 Syndication

RDF Site Summary 혹은 Really Simple Syndication의 약어인 RSS라는 기술이 빠르게 전파되고 있다. RSS는 웹사이트에서 최근 수정을 반영하는 웹에서 파일들을 위한 잘 정의된 문법이다. 그런 동안 피드를 위한 다른 형식이 개발되었고 Atom이나 Kllip가 사용된다.

syndication feed를 종종 제공하는 사이트의 예들은 다음과 같다.

- Weblogs

- News sites
- Special topics
- Weather forecasts
- Picture sites

이것은 웹 사용자가 최신의 것을 갖는 방법을, pull-method에서 push-method로 바꾼다. 사용자는 뉴스가 있는지 페이지를 보면서 진행하지 않고, 그의 RSS 리더가 피드(RSS 파일)을 정기적으로 요청할 때 뉴스가 그에게 푸시된다.

물론 이러한 콘텐츠 syndication은 stand-alone RSS리더들을 통해서만 사용되어 질 수 없고, MyOpera Community Blogs나 구글 개인화 탐색 페이지에서 웹 로그 sidebar 처럼 웹 페이지에 통합된다. 나중에 syndication은 다양한 서비스들의 통합의 수단으로 사용되어진다.

개인화 뉴스 syndication은 웹의 초창기에 선 보였다. 한 예가 Linux KDE manager를 위한 KNewsTicker이다.

4.3 RDF

RDF(Resource Description Framework)는 모든 타입의 자원들을 기술하기 위한 명세이다. 그것은 형식적으로 XML의 sub-언어로서 명시하지만, 그것의 데이터 모델은 여러 방법들로 표현되어 질 수 있다. 해당되는 schema 언어인 RDFS와 같이 소위 말하는 어떤 자원들의 영역을 위한 ontologies는 모델될 수 있고 형식적으로 기술되어질 수 있다.

두 종류의 잘 알려진 RDF ontologies는

- Dublin Core Metadata Initiative의 어휘, 예를 들면 HTML <meta> 요소들의 표현을 확장하기 위해서 HTML 헤더에서 사용되어진다.
- 초기 버전에서 RDF로 공식적으로 표현되어진 RSS 문법

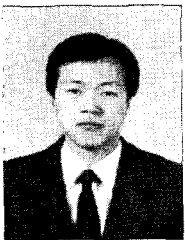
RDF(S)는 시맨틱 웹의 언어로 간주되지만 Web 2.0와 같이 언급되어진다.

5. 결론

2.0이라는 버전은 서비스 제공자나 이용자 입장에서 좀 더 나은 환경을 제공해야 한다는 것을 의미한다. 즉 Web 2.0이 좀 더 나은 웹을 서비스하기 위해 콘텐츠 생산력을 강화하는 방법을 고려해야 하듯이 Web 2.0 기술들도 좀 더 나은 웹을 개발하기 위해 무엇을 할 수 있는지 이해해야한다.

Web 2.0 기술은 클라이언트 기술과 서버 기술을 광범위하게 포함하고 있다. 클라이언트 기술은 주로 사용자에게 풍부하고 강력한 사용자 인터페이스를 제공하며, 서버 기술은 서비스가 가지고 있는 데이터를 어떻게 쉽고 다양하게 클라이언트에게 제공하는지를 중점적으로 다룬다. 이러한 Web 2.0 기술은 Web 2.0에 대한 충분한 이해를 바탕으로, 기술 중심이 아니라 사람 중심의 서비스라는 점을 망각하지 않는다면 사용자들에게 사랑받는 Web 2.0 서비스를 만들 수가 있다.

◎ 저 자 소 개 ◎



한 명 목

- 1976~1980 연세대 공과대학 졸
- 1985~1987 뉴욕공대 대학원졸
- 1992~1997 오사카시립대학원졸(공학박사)
- 1998~ 경원대학교 소프트웨어학부 부교수
- 2005~2006 조지아공대 교환교수