

# 폴링 기법을 활용한 실시간 웹 프레젠테이션

천성규\*, 이원규\*\*

\*고려대학교 컴퓨터교육학과

\*\*고려대학교 정보대학 컴퓨터학과

e-mail:sungkyu.chun@inc.korea.ac.kr

## Real time Web Presentation using Polling method

SungKyu Chun\*, WonGyu Lee\*\*

Dept. of Computer Science Education, Graduate School, Korea University

Dept. of Computer Science and Engineering, College of Informatics, Korea University

### 요 약

웹 응용 프로그램은 운영체제에서 동작하는 응용프로그램에 비해 사용의 범위와 양이 폭발적으로 증가하는 추세이다. 이러한 흐름에 맞추어 다양한 웹 응용프로그램이 개발되고 있고, 다중의 사용자가 동일한 페이지를 볼 수 있는 실시간 웹 프레젠테이션 프로그램을 폴링 기법을 활용하여 제작되고 있다. 하지만 이러한 문제가 있어 비동기 데이터의 순서제어에 관한 연구가 필요하다. 본 연구에서는 이러한 문제를 해결하기 위해 비동기 데이터의 순서 제어에 대해 연구하였고, 타임스탬프 기법을 활용하여 해결하고자 하였다. 타임스탬프 기법을 활용하여 클라이언트와 서버는 각각 결과의 순서와 요청의 순서를 제어하여 불필요한 통신 대역폭 낭비를 줄였으며, 순차적으로 진행 되어야 하는 프레젠테이션 응용 프로그램의 특성에 맞는 결과를 보일 수 있게 되었다.

### 1. 서론

웹 응용프로그램은 wifi 모듈이 장착된 모바일 장비인 스마트 폰의 보급과 함께 폭발적으로 그 영역과 활용을 지난 몇 년간 증대시켜 왔다. 이러한 상황아래 특히 사무를 위한 프로그램이 대형 포털과 검색엔진 제공자로부터 발매되어 많은 사용자가 사용하고 있는데, 저작도구로서의 기능에 초점이 맞추어져 있는 것이 현실이다.

이 논문에서는 사무를 위한 웹 응용 프로그램 중, 저작 외의 공유 기능이 함께 포함 되어야 하는 프레젠테이션 프로그램에 주목하여, 실시간으로 프레젠테이션 페이지를 간단한 웹 주소 접근만으로 공유 할 수 있도록 하며 나아가 발표자가 활용 할 수 있는 페이지 넘김, 페이지 이동 등의 기능을 함께 제공하는 웹 응용 프로그램을 소개한다.

이 웹 응용 프로그램은 폴링 기법을 활용하여 제작되었으며, 이때 발생하는 사용자측의 비동기적 데이터 수신을 처리하기 위해 사용한 기법에 대해 설명한다.

응용 프로그램에서 크게 고려되는 사항은 아니다. 그러나 예상치 못한 통신 실패 및 재시도로 인해 발생한 지연은 순차성이 필요한 데이터를 제어할 때 반드시 고려해야 하는 부분이다. 아래 (그림 1)은 각종 지연을 포함한 웹 응용 프로그램의 실시간성 개념을 표현한 도식이다.



(그림 1) 실시간 웹 응용 프로그램에서 고려해야 할 지연의 종류

### 2. 관련연구

#### 2.1 웹 응용 프로그램에서의 실시간

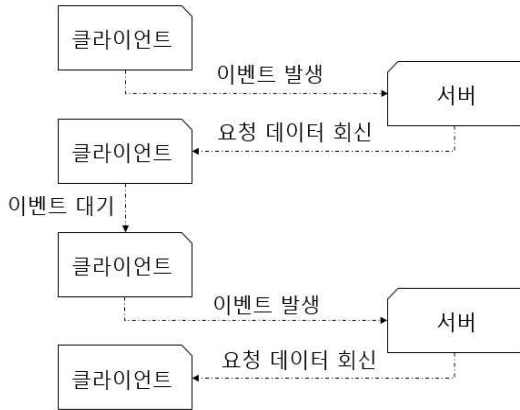
##### 2.1.1 웹 응용 프로그램에서 실시간의 개념 및 특성

웹 어플리케이션에서 사용자에게 제공하는 실시간성은 통신상의 왕복 수신 시간을 포함하는 개념이 된다. 충분한 대역폭이 제공 될 경우 왕복 수신 시간은 일반적으로 클라이언트, 혹은 서버의 처리 시간보다 짧으며 웹 실시간

#### 2.2 실시간 웹 응용 프로그램의 구조

##### 2.2.1 폴 방식

폴 방식은 사용자가 발생시키는 이벤트를 바탕으로 웹 응용 프로그램의 연산을 담당할 서버와 실시간 데이터 동일성을 유지하기 위한 여러 수단 중 하나이다.[1]



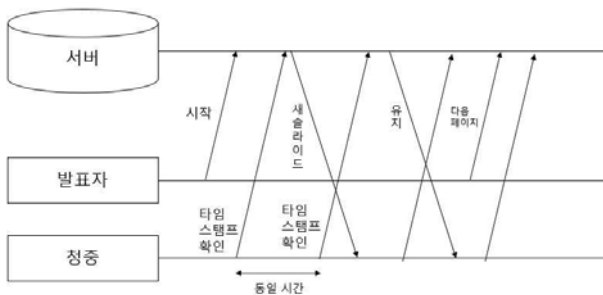
(그림 2) 풀 방식의 동작 예제

(그림 2)의 흐름과 마찬가지로, 풀 방식은 일반적으로 클라이언트에서 발생하는 이벤트에 대한 응답을 서버가 대응하는 구조로 이루어져 있다. 풀 방식의 단점으로는 지속적인 요청과 회신으로 인한 데이터 전송 대역폭의 낭비가 있으나, 구현의 단순함과 응용 프로그램 구조를 간단히 할 수 있어 선호된다[2].

이 논문에서 우리는 풀 방식을 채택하되, 풀방식이 가지는 단점을 극복하기 위해 단위 시간당 이벤트를 클라이언트가 발생 시키되, 서버측의 데이터 갱신을 클라이언트가 가진 최종 변경 타임 스탬프 값과 비교 후 그 값이 다를 경우에만 실제 데이터를 풀 방식으로 요청하는 트릭을 사용 하였다[3].

### 3. 실시간 프레젠테이션 웹 응용 프로그램의 구현

#### 3.1 프레젠테이션 웹 응용 프로그램의 구조



(그림 3) 호출 순서 구조

(그림 3)은 우리가 제시하는 프레젠테이션 웹 응용 프로그램의 통신 구조이다. 발표자가 시작 시그널을 서버에 입력 하는 것으로 시작하여, 청중의 타임스탬프는 서버의 타임스탬프와 불일치하게 된다. 이후 청중 쪽 폴링 이벤트가 시간에 따라 발생하게 되면, 새로운 슬라이드를 서버가 전송하게 된다. 이때, 서버는 클라이언트가 전송하는 타임

스탬프를 기반으로 하여 간단한 유지 시그널과, 새로운 슬라이드 파일 중 전송 할 것을 선택 하게 된다.

타임 스탬프에 기반한 방법은 폴링으로 인해 발생하는 데이터 전송 대역의 낭비를 최소화 하는 효과가 있다.

우리는 이때, 시간 간격을 평균 통신키지연 2초를 포함하여 약 5초에 1회 폴링을 발생 시키도록 설정 하였다. 이러한 설정은 페이지 이동이 발생 하였을 시 지연을 감안 하여도 충분히 실시간으로 청중들의 장비에 현재의 페이지로 이동 하게 할 필요가 있었기 때문이다.

### 3.2 클라이언트와 서버의 동작 알고리즘

#### 3.2.1 클라이언트의 동작 알고리즘

<표 1> 클라이언트 동작 알고리즘

```

1 client_time_stamp = 0;
2 client_page_idx = 1;
3
4 start
5     compare_timestamp();
6     while(1){
7         wait(5second);
8         compare_timestamp();
9     }
10 end
11
12 start compare_timestamp(){
13     e = ajax_get_server_time_stamp();
14     if(client_time_stamp => e){
15         do nothing
16     }else{
17         pull_slide();
18         client_time_stamp = e;
19     }
20 }end compare_timestamp;
21
22 start pull_slide(){
23     slide[] = ajax_get_slide();
24     for(;client_page_idx<= slide.length;
25         ++client_page_idx){
26         DOM.add(slide[client_page_idx]);
27     }
28 }end pull_slide;

```

클라이언트의 동작은 초기화 이후 크게 두 가지 동작으로 구분된다.

첫째로, 클라이언트는 현재 자신이 가진 타임스탬프를 서버에 5초 단위로 요청하는 타임스탬프와 서버의 타임스탬프를 비교 하기 위해, 지속적으로 서버의 타임스탬프를 요청, 이후 비교 하는 부분이다.(라인 12~20)

둘째로, 서버와 자신의 타임스탬프가 다르며, 서버의

타임스탬프가 클라이언트 자신이 가진 타임스탬프 보다 이후의 것일 경우 동작하는 슬라이드를 가져와서 DOM Tree의 새로운 마디로 추가하는 부분이다.(라인 22~27)

이러한 클라이언트의 동작을 처리하기 위해 지원되어야 할 서버의 기능을 다음 단락에서 살펴본다.

### 3.2.2 서버의 동작 알고리즘

<표 2> 서버의 타임스탬프 처리 알고리즘

```

1 con = db_connect();
2
3 sever_time_tamp = get_time_stamp(con);
4
5 die(sever_time_tamp);
    
```

<표 3> 서버의 슬라이드 요청 처리 알고리즘

```

1 con = db_connect();
2 slide[] = array();
3
4 pages = get_page_idx(con);
5 for(count = 1; count <= page; count++){
6 slide[count][src = slide_data(count, con)
7 }
8 slide[0] = pages
9 die(slide);
    
```

서버의 동작은 타임스탬프의 요청을 받아 클라이언트에게 전송하는 부분과, 클라이언트의 슬라이드 요청을 받아 처리 하는 부분으로 나뉜다.

서버에서 타임스탬프를 제공해 주더라도 클라이언트에게 전송 성공여부는 보장 될 수 없으며, 그 도착 순서도 전송한 순서와 다를 수 있으므로, 서버에서 슬라이드를 제공하는 로직은 보다 섬세한 데이터 패키징을 필요로 한다. 즉, 서버에서 슬라이드를 클라이언트에게 보낼 때는 클라이언트가 자신이 받은 슬라이드가 어느 시점에 보낸 것이며, 자신이 현재 가지고 있는 것보다 전에 요청한 데이터가 뒤늦게 도착한 것이 아닌지 판단 할 수 있어야 한다.

이런 부분을 보장해 주기 위해 슬라이드 데이터의 0번 슬롯은 이후의 슬라이드 데이터들과는 다르게 별도의 페이지값으로 되어 있으며, 이는 클라이언트 측에서 서버가 보낸 데이터의 발송 순서를 확인 할 수 있도록 해 준다.

## 4. 결론 및 제언

우리는 간략화된 웹 프레젠테이션 응용 프로그램의 동작과 그 구조 설계를 살펴 보았다. 더불어 시스템을 구현하기 위한 기초적인 기술인 폴링과 폴링의 단점인 통신

대역폭 낭비를 축소 하기 위한 타임스탬프 기법 역시 확인하였다.

이러한 구조로 만들어진 웹 프레젠테이션 응용 프로그램의 경우 실시간적으로 청중과 발표자가 동일한 화면을 공유 할 수 있다는 장점이 있으며, 이는 청중의 태도 변화를 이끌어 낼 수 있는 요소라고 추측되어진다[4]. 이런 추측을 뒷받침 하기 위해서는 후속 연구 과제로 이 응용 프로그램을 활용한 발표, 혹은 수업 등을 실제 구성하여 사전 사후 통제에 따른 수용자의 반응 차이의 유의미성을 확인 할 필요가 있을 것으로 보인다.

## 참고문헌

- [1] 임준호(2011). COMET을 이용한 실시간 웹 어플리케이션. 석사학위 논문, 고려대학교.
- [2] 김훈외 2인(2009), RIA를 적용한 무선 센서네트워크의 실시간 메시지 시스템. 한국HCI학회 2009년 학술대회 논문지
- [3] 김종근외 3인(2012), 실시간 웹 게시판 모니터링 및 모바일웹을 이용한 알람 서비스 개발, 학술논문, 디지털 콘텐츠학회지 13권 제 1호
- [4] 최혜진(2015), 실시간 웹 토론을 통한 학생들의 수학적 의사소통과 수학적 태도 분석, 석사학위 논문, 한국교원대학교