

BioPlace의 일정관리 이벤트 처리

박희종^o 문남두 안건태 박양수 이명준

울산대학교 컴퓨터·정보통신공학부

{phj34^o dooya java2u yspark mjlee}@mail.ulsan.ac.kr

Event Handling of Schedule Management System in BioPlace

Hee-Jong Park^o, Nam-Doo Moon, Geon-Tae Ahn, Yang-Su Park Myung-Joon Lee
School of Computer Engineering & Information Technology, University of Ulsan

요 약

BioPlace의 일정관리 시스템은 특정 연구 분야의 연구자들 사이에 효과적이고 능률적인 일정관리를 지원하기 위한 웹기반 환경이다. 웹을 통한 공동작업의 경우 개인은 물론 그룹의 일정을 보다 효율적이고 체계적으로 수립하고 공지하는 것이 바람직하다. 본 논문에서는 이러한 요구를 충족시키기 위하여 BioPlace의 일정관리 작업에서 발생하는 이벤트의 흐름과 관련 데이터의 분석을 통한 데이터의 효율적인 관리방법을 기술하였다. 일정관리 작업에 사용되는 데이터의 효율적인 관리와 처리를 위한 방법으로 세션과 Request 객체를 이용하였다.

1. 서 론

BioPlace 시스템은 유전체 연구자들 사이에 효과적으로 연구에 대한 정보를 교환하고 유전체 연구와 관련된 서비스를 이용할 수 있도록 지원하는 웹기반 협업지원 환경이다[1]. 그리고 BioPlace의 일정관리 시스템은 같은 그룹에 속한 팀원들이 사용자 중심의 인터페이스를 이용하여 효과적으로 일정을 관리할 수 있는 환경이다. BioPlace 시스템에서 만들어진 팀은 프로젝트의 원활한 진행을 위해 수립된 일정을 팀원들에게 알려야 한다. 팀장은 이 시스템을 이용하여 팀의 일정을 추가, 수정, 삭제 등을 할 수 있다[2,3].

이러한 일정관리 시스템은 서로 의존적으로 동작을 하는 두개의 프레임으로 나뉘어져 있다. 즉, 날짜의 변환을 편리하게 하고 월간일정, 주간일정을 보여주기 위한 메뉴가 위치한 상위 프레임과 이 메뉴에 대한 내용과 상세 월간일정을 보여주는 기능 및 일정추가, 특정일정보기, 하루일정보기 등의 다양한 기능을 하는 하위 프레임으로 나뉘어져 있다[4].

본 논문에서는 편리하고 효과적인 공동작업의 수행을 위해 일정관리 시스템이 프레임간의 이벤트에 대한 데이터가 어떤 것들이 있고, 이 데이터들이 어떻게 전달되고 어떻게 처리가 되는지에 대해 기술하려고 한다. BioPlace의 일정관리 시스템은 현재 플랫폼 독립적인 자바스크립트 및 JSP/EJB 기술[5]을 기반으로 작성되었고 효과적이고 편리한 일정관리를 위하여 사용자 중심의 다양한 기능을 제공한다.

본 논문은 다음과 같이 구성되었다. 서론에 이어 2장에서는 일정관리 시스템에 필요한 데이터들이 어떤 것들이 있고 어떻게 저장되는지에 대해서 설명하고 3장은 일정관리 시스템의 이벤트와 그에 따른 데이터 처리 방법에 대해 기술한다. 마지막으로 4장에서는 결론을 기술하고자 한다.

2. 일정관리 시스템에 필요한 데이터들

* 본 연구는 정보통신부지원 "정보통신기초기술연구지원 사업"의 지원으로 이루어졌음

일정관리 시스템에서 가장 많이 사용되고 중요한 데이터는 일정에 대한 날짜정보이다. 어떤 프레임에서 날짜의 변환이 일어났는지 정확한 파악이 된다면 프레임간의 날짜에 대한 동기화 및 관리가 쉬워진다. 프레임간의 데이터에 대한 동기화와 관리를 위해 데이터들을 세션에 저장하거나 Request 객체를 이용한다. 그래서 사용자가 클릭하여 특정한 날짜로 변환이 될 때마다 년, 월, 일에 대한 값들이 항상 세션에 저장된다. 그리고 개인일정, 팀 일정, 전체일정을 구분해 주는 모드, 상황에 맞는 데이터를 읽어오기 위해 사용되는 플래그에 대한 값들도 세션에 저장된다. 이 값들은 모든 프레임에서 언제나 필요한 값들이기 때문에 세션에 저장되어 편리하게 사용되고 관리된다. 그렇지만 이렇게 세션에 저장된 값들을 사용하기 위해 항상 세션에 접근하여 값을 가져와 사용하는 것은 아니다. 세션의 접근횟수를 줄이기 위해 세션에 있는 데이터의 값이 필요한 경우라도 상황에 따라 인자로 간단히 넘겨줄 수 있는 데이터는 Request 객체로 받아서 사용한다.

[표 1] 세션 또는 Request 객체를 이용하는 변수와 기능

구분	변수명	기능
세션	D_YEAR	년도를 세션에 저장
	D_MONTH	달을 세션에 저장
	D_DAY	날을 세션에 저장
	D_MODE	모드를 세션에 저장 (0=전체, 1=개인, 2=팀)
	D_FLAG	상황에 맞는 데이터를 읽어오기 위한 플래그
Request 객체	id	일정에 대한 고유 id를 저장
	user_id	사용자에 대한 고유 id를 저장
	user_name	사용자의 이름을 저장

일정관리 시스템에서는 세션에 저장되는 값들 외에도 계획된 일정에 대한 고유 id, 사용자에 대한 고유

id(user_id), 그리고 사용자 이름(user_name)과 같은 데이터들이 필요하다. 이 값들은 세션을 이용하지 않고 직접 인자로 넘겨주고 인자를 넘겨받은 프레임에서는 Request 객체로 받아서 적절하게 사용한다.

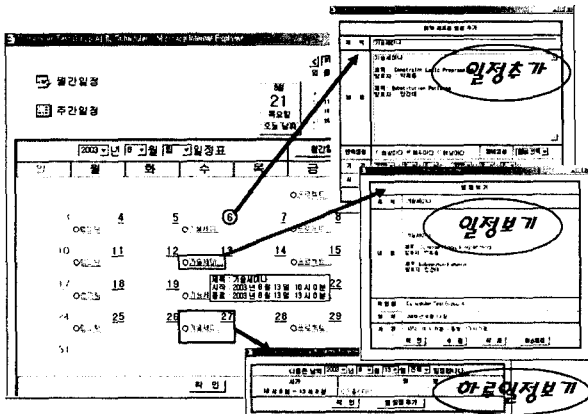
이렇게 세션을 이용하여 관리되고 사용되는 값들과 세션을 이용하지 않고 사용할 수 있는 값들을 분리하여 사용함으로써 시스템에 미치는 영향을 최소화할 수 있다. [표 1]은 세션과 Request 객체를 이용하는 변수와 그 기능에 대해서 설명하고 있다.

3. 일정관리 시스템의 이벤트 처리

일정관리 시스템에서 발생하는 각 프레임간의 이벤트 처리를 위한 데이터는 세션을 이용하는 방법과 인자로 넘겨온 값을 Request 객체를 이용하는 두 가지 방법을 이용하였다. 전자는 세션에 넣어놓은 값은 어느 프레임에서나 사용이 가능하지만 세션을 너무 많이 사용하면 메모리의 오버헤드가 발생하는 단점이 있다. 후자의 경우는 프레임에서 인자로 넘겨준 값은 다음 프레임이나 특정 프레임에서만 사용이 가능하다는 단점이 있다. 하지만 메모리를 사용하지 않고 값을 직접 로드하려고 하는 프레임에 전달해 주기 때문에 시스템의 부하를 줄일 수 있다.

이렇듯 두 방법은 모두 장단점을 가지고 있기 때문에 이 방법들을 적절하고 유용하게 장단점을 살려 적용하는 것은 중요하다.

3.1 이벤트의 종류



[그림 1] 메인 인터페이스에서 발생하는 이벤트의 종류

메인 인터페이스에서 발생하는 이벤트는 [그림 1]과 같이 셀, 날짜 그리고 특정일정을 클릭했을 때 발생하는 세 종류의 이벤트가 있다. 날짜와 특정일정에 대한 클릭 이벤트는 모두 셀 위에 있기 때문에 이 두가지의 이벤트가 발생했을 때는 항상 셀에 대한 이벤트는 발생하게 된다. 이러한 이벤트에 대한 문제 때문에 [그림 2]와 같은 방법을 이용하여 메인 인터페이스에서 발생하는 이벤트를 효과적으로 분리하였다.

```

<셀을 클릭했을 때 (하루일정 보기)>
onclick = if(event.srcElement.tagName=='TD')
           window.open('d_scd_list.jsp? --생략--

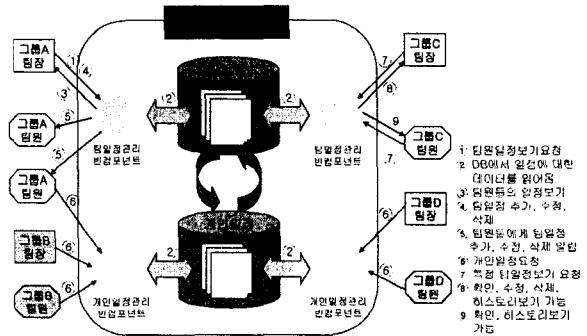
<날짜를 클릭했을 때 (일정추가)>
onclick="<% int GroupMaster =
           workGroup.getGroupMaster(team_id);
           if(GroupMaster == user_id){ %>
           window.open ('scd_input.jsp? --생략--

<특정일정을 클릭했을 때 (특정일정 보기)>
onclick="window.open('scd_view.jsp? --생략--
    
```

[그림 2] 메인 인터페이스에서의 이벤트 분리방법

날짜나 특정일정을 클릭했을 때 셀에 대한 이벤트가 발생하는 것을 방지하기 위해 클릭에 대한 이벤트가 셀에 대한 이벤트인지를 파악하여 만약 그 이벤트가 셀에 대한 이벤트가 아니라면 특정일정이나 날짜에 대한 이벤트를 발생시킨다. 그것을 파악하는 방법은 클릭이벤트가 'TD' 태그에 대한 이벤트인지를 검사함으로써 세 개의 이벤트를 구분할 수 있다. 이벤트에 대한 구분이 되었다면 일정관리 시스템은 결정된 이벤트에 적절한 프레임을 로드한다.

3.2 작업장내의 이벤트 흐름



[그림 3] 작업장내의 개인 또는 팀일정에 대한 이벤트 흐름

팀작업을 위해 일정을 추가할 때는 먼저 팀장이 일정을 추가하기 위한 이벤트를 발생시키면 일정추가 프레임이 로드된다. 일정추가 프레임에서는 일정추가를 위해 팀원들의 일정확인을 원할 때는 팀원의 일정을 확인한 뒤 적당한 날짜와 시간에 일정을 추가한다. 추가된 일정을 빈 컴포넌트에 의해서 팀일정 데이터베이스에 저장되어 관리된다. 이렇게 수립된 일정은 팀에 속한 모든 팀원들에게 일정이 수립되었음을 알린다.

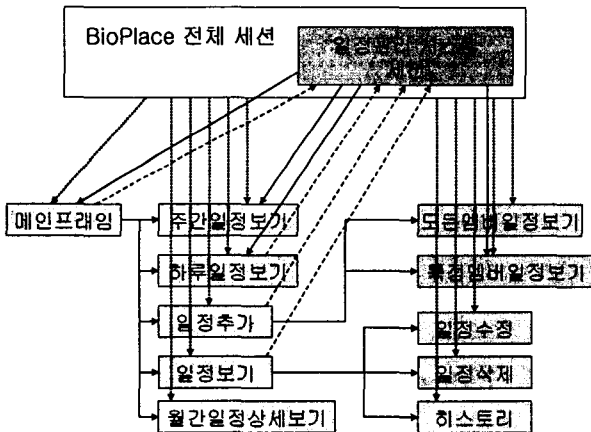
일정추가에 의해 이미 수립된 일정을 수정하거나 삭제하기 위해서는 일정을 확인하는 이벤트에 의해 로드된 프레임에서 수정 또는 삭제 이벤트에 의해서 일정이 처리된다. 이 과정도 수정되었거나 삭제된 일정은 데이터

베이스에 새로 저장되거나 삭제되고 일정이 수정 또는 삭제되었다는 내용을 모든 팀원들에게 알려준다.

개인일정에 대한 추가, 수정, 삭제에 대한 이벤트의 흐름은 팀일정에 대한 과정과 거의 비슷하다. 단지 개인일정이기 때문에 다른 사람들에게 일정에 대한 알리를 하지 않는다는 차이뿐이다. [그림 3]은 작업장내에서 개인 또는 팀일정에 대한 이벤트 흐름에 대해 나타내고 있다.

3.3 이벤트 처리

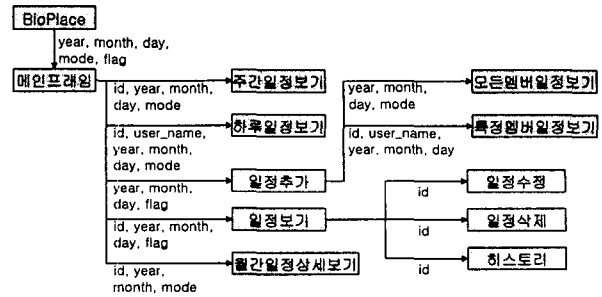
일정관리 시스템에서 프레임간에 발생한 이벤트에 대해 필요한 데이터는 세션과 Request 객체에 의해서 전달된다. [그림 4]은 세션으로 처리되는 데이터의 흐름을 나타내고 있다. 일정관리 시스템에서는 각각의 기능을 수행하기 위해 SharedScheduleHome, PersonSchedule, UserInfoHome, workGroupHome, Uid, Gid등과 같은 BioPlace 시스템에서 이미 만들어져 세션에 저장되어 있는 변수들과 특별히 일정관리 시스템에서만 사용하기 위해 만들어진 D_YEAR, D_MONTH, D_DAY, D_MODE, D_FLAG등의 변수들을 정의하고 있다. 이 변수들은 필요할 때 언제든지 세션에서 데이터에 가져와 사용할 수 있다. 이렇게 함으로써 어느 프레임에서든지 같이 사용할 수 있는 변수에 공통의 값을 공유하여 사용할 수 있다.



[그림 4] 세션으로 처리하는 데이터의 흐름

이렇듯 전체적인 데이터의 흐름에 대한 처리는 두 가지의 방법을 이용함으로써 조금이라도 더 효율적인 데이터 관리를 할 수 있다.

그러나 모든 프레임에서 사용하지 않고 특정한 프레임에서만 사용을 하는 변수에 대해서는 인자로 보내어 Request 객체로 받아서 사용한다. [그림 5]는 프레임간에 인자로 값을 보내고 그 값을 Request 객체로 가져와 처리하는 데이터의 흐름을 나타낸다. 특별히 세션에서 관리를 하지 않아도 되는 id(일정에 대한 id), user_id, user_name과 다음 프레임을 로드하기 위해 사용되는 year, month, day, mode, flag등이 인자로 넘겨지고 Request 객체로 받아서 유용하게 사용한다.



[그림 5] request로 처리하는 데이터의 흐름

다중사용자 환경의 경우 세션을 많이 이용할수록 서버 부하가 높아진다. 서버부하가 높으면 당연히 일정관리를 위해 소요되는 시간이 늘어난다. 그래서 서버부하를 낮추기 위해 모든 데이터를 세션에 저장하지 않고 꼭 세션에 저장해야만 하는 최소한의 데이터에 대해서 세션에 저장하고 나머지는 Request 객체를 이용하고 있다. 이렇게 서버의 부하를 줄임으로서 일정관리 작업에 대한 시간을 줄일 수 있다.

4. 결론

본 논문은 BioPlace의 일정관리 시스템을 편리하고 효과적으로 이용하기 위한 이벤트와 그 이벤트로 발생된 데이터의 처리 방법에 대해서 기술하였다. 모든 이벤트에 대하여 어느 프레임에서나 참조되어야 하는 데이터는 세션을 이용한 방법을, 다음 프레임에서만 필요한 데이터는 인자로 전달하여 Request 객체로 받는 방법을 선택하였다. 따라서 이렇게 두 방법을 병행하여 사용함으로써 어느 프레임에서나 정확한 데이터를 유지하며 일정을 관리할 수 있다. 또한 많은 세션의 사용을 최소화함으로써 다중사용자 환경에서 발생할 수 있는 서버부하를 줄이도록 시스템을 설계 및 구현하였다.

[참고문헌]

- [1] Myung-Joon Lee, Chun-Yong Han, Geon-Tae Ahn, Jin-Hong Kim, Nam-Doo Moon, Myung-Hee Jung, "CoWare : A Web-based Groupware for Effective Collaboration", Proceedings The 4th Korea-Russia International Symposium on Science and Technology, Part3 Machine Parts and Materials Processing, June 27-July 1, 2000 at the University of Ulsan, Republic of Korea, p.128-133, 2000.
- [2] 안건태, 정명희, 이근용, 윤남두, 이명준, "iPlace:EJB 기술을 이용한 웹 기반 협업시스템", 정보처리학회논문지 D 제 8-D권 제 6호(2001년 12월)
- [3] 정명희, 안건태, 윤남두, 김진홍, 이명준 "웹 기반 공동작업공간의 지원을 위한 EJB 컴포넌트의 개발" 한국정보과학회 '2001 봄 학술발표논문집(B) 제 28권 1호, p.469-471, 2001
- [4] 박희중, 윤남두, 안건태, 이명준, "BioPlace 협업지원 시스템에서의 일정관리", 한국정보과학회 '2003 봄 학술발표논문집(A) 제 권 호, p. - , 2003
- [5] EJB™ Tech "Server Component Model for the Java™ Platform" by Anne Thomas, December 1998, Prepared for Sun Microsystems, Inc.