

IBM Research Report

Event Maps: A Collaborative Calendaring System for Navigating Large-Scale Events

Jingtao Wang*

Computer Science Division
UC Berkeley
387 Soda Hall
Berkeley, CA USA

Danny Soroker, Chandra Narayanaswami

IBM Research Division
Thomas J. Watson Research Center
P.O. Box 704
Yorktown Heights, NY 10598 USA

* Work predominantly done while an intern at IBM TJ Watson Research Center, Hawthorne, NY 10532



Research Division

Almaden - Austin - Beijing - Cambridge - Haifa - India - T. J. Watson - Tokyo - Zurich

Event Maps: A Collaborative Calendaring System for Navigating Large-Scale Events

Jingtao Wang ^{♦§*}

♦Computer Science Division
UC Berkeley, 387 Soda Hall, Berkeley, CA, USA
jingtaow@cs.berkeley.edu

Danny Soroker [§]

§IBM T.J. Watson Research Center
19 Skyline Dr, Hawthorne, NY, USA
{soroker, chandras}@us.ibm.com

Chandra Narayanaswami [§]

ABSTRACT

Event Maps is a novel, rich and interactive web-based system targeted at improving the experience of attending and organizing large, multi-track conferences. Through its zoomable *Tabular Timeline*, users can navigate the conference schedule, seamlessly moving between global and local views. Event Maps enables contextual asynchronous collaboration before, during, and after the conference. Organizers can easily create or import conference schedules via a backend interface, and also use the provided analytic toolkits to get insights from visiting patterns and statistics. Event Maps has gone through three major rounds of iterative design, including a deployment at an intra-company conference. This paper describes in detail the design and evolution of Event Maps, including the quantitative and qualitative feedback from our real world deployment and the results of a comparative study with a traditional web-based conference calendar.

Author Keywords

Temporal Data, Information Visualization, Tree, Collaborative Visualization.

ACM Classification Keywords

H.5.2 [Information Interfaces and Presentation]: User Interfaces; I.3.6 [Methodology and Techniques]: Interaction Techniques.

INTRODUCTION

Attending conferences and trade shows is important to our professional and social life [22]. Large conferences often have many concurrent sessions, and it may be challenging to get a good sense of all available activities and plan one's attendance. Having a guiding *map* can help tremendously.

Online schedules on traditional conference web sites [8] are

* Work predominantly done while an intern at IBM TJ Watson Research Center, Hawthorne, NY 10532.

built as a static collection of hyperlinked pages, where users may have to click multiple times and go back and forth to locate the information they need. Such a design makes it difficult to simultaneously explore both depth and breadth, and the user loses context rapidly. It is also hard to create tailored views of the schedule according to your preference.

Exchanging information with other conference participants, a key objective while attending conferences, is also not supported by this design. Although many conferences do provide complementary blog/wiki systems, they are generic to the conference and not tightly coupled to specific sessions. Why can't comments of senior researchers on an award winning paper be made available to new students?

Finally, according to our informal survey, creating the schedule on a conference web site and keeping it up-to-date is a challenging, tedious and error-prone task for conference organizers with present day systems.

Event Maps is a rich internet application we have built to address these shortcomings in navigation, collaboration, personalization, and organization.

Event Maps aims to make the experience of interacting with an online conference calendar engaging, productive and intuitive. It supports seamless navigation between broad and deep views of the schedule information, asynchronous interaction in the context of individual sessions, and the ability to tailor preferences and favorites. As such, it sets up the conference calendar as a focal point for interaction and data mining for participants, organizers, and any other interested parties, which has value before, during and after the conference is held.

The main research contributions of the work described here are in devising and exploring the use of:

- *Tabular Timeline*: A zoomable calendar interface for large event navigation to minimize page switching.
- *Active Corners*: Compact decoration widgets for awareness and activation of features.

We provide tangible evidence that, aided by these mechanisms, Event Maps indeed supports task efficiency while being enjoyable to use.

By adhering to the iCalendar standard [21] for calendar information, Event Maps makes it easier for conference organizers and administrators to maintain an updated conference schedule. The system also provides analytic administrative tools that help get insights from visiting patterns and statistics.

The rest of the paper is organized as follows. After discussing related work, we describe the Event Maps system, its evolution and implementation. We then describe the user studies and deployments to date, followed by a discussion of lessons learned, future work, and conclusion.

RELATED WORK

Research in group/collaborative calendaring systems has explored topics such as scheduling a group activity based on the availability of each team member, ambiguous time constraints [5] or special audiences such as family members [17]. However, less research has been devoted to addressing the needs of conference participants.

The idea of using a zoomable interface (ZUI) to view calendars is not new. In 1991, Furnas [10] proposed a textual Lisp-based calendar program with fisheye distortion. Bederson et al. [3] later extended this idea and built a scalable visual fisheye calendar named DateLens. Event Maps differs from DateLens in several ways: First, DateLens focuses on *personal events* while Event Maps is targeted at *multi-track conference events* with selective expansion of more than one track. Second, DateLens is a single user application while Event Maps supports asynchronous collaboration in the context of conference sessions. Finally, as DateLens is not a web-based application, it lacks facilities for user tracking and data analysis to help administrators and system developers.

Event Maps deals with temporal data [14]. Visualization systems for temporal data have been available for decades, supporting activities such as abnormality detection, record management [1] and decision making. LifeLines [17] presents a timeline style interface to manage personal medical record and supports zooming, panning and text filtering. TimeSearcher [13] lets users select stock price patterns of interest via a direct manipulation interface named TimeBox that filters through thousands of temporal records of numerical data.

Several pieces of work address the scalability problems encountered when visualizing temporal data [1, 4, 5, 13, 25]. In addition to data level transformations [15], most of the solutions are domain specific and fall into the following two categories: 1. Using a zoomable interface to highlight the region of interests when necessary [4, 5]. 2. Designing a query interface, either by setting up parameters via on-screen data widgets such as sliders, or by using techniques such as direct manipulation or query by example to refine targets of interests [1, 13, 17].

Recently, we have seen the emergence of collaborative visualization systems like Many Eyes [26], sense.us [12],

and Swivel [23], which support creating visualizations from statistical data, and also provide text comments, tagging and view-sharing through book-marking. Systems that partition work across time and space have been demonstrated to be beneficial for group activities [11]. Our work extends some of these principles to multi-track conference schedules.

THE EVENT MAPS SYSTEM

The major features of the Event Maps system include:

- A zoomable interface that allows quick transition between views at different levels of detail, and simultaneous viewing of multiple levels of detail.
- In-place support for asynchronous collaboration features such as highlighting, and commenting directly in the context of conference sessions.
- Functions for conference organizers to create and import conference schedules, keep them updated, and obtain insights from visiting patterns and statistics.

A big challenge in designing Event Maps has been to support the rich feature-set in a clean, intuitive and efficient interface. Quick event navigation and location is supported via a zoomable interface coupled with features such as progressive searching, brushing [16], and details on demand [6]. Contextual personalization and collaboration are supported via compact decoration widgets. Event Maps encourages spontaneous interface exploration by mapping frequent operations to mouse hovering clicking.

User Interface

The primary interface for Event Maps is shown in Figure 1. In the center region is a zoomable widget named *Tabular Timeline* to display all the activities in a conference. The top panel provides a button to set the *Tabular Timeline* to the original state, a search box (Figure 1.f), widgets for selecting the current view and time zone, and color-coded session-category labels. The bottom panel is an area for managing the user's "favorite" sessions; sessions with time conflicts will be marked in red in the "My Favorites" region. The rectangles inside the *Tabular Timeline* are conference sessions, color-coded by their associated categories. The top-left and top-right corners of each session are called *Active Corners*, and are used to support personalization and asynchronous.

Tabular Timeline

Most Event Maps interactions happen inside the *Tabular Timeline*. The initial view of *Tabular Timeline* is an overview of the entire conference, with each of the tracks¹ collapsed. Our *Tabular Timeline* widget is a combination of traditional calendar visualization and a zoomable

¹ The definition of track depends on the conference: it can either be a theme including sessions happening in multiple locations at the same time (e.g., Lotusphere 2009), or sessions in a single location (e.g. CHI 2009). Event Maps supports both definitions.



Figure 1. The Event Maps System. (a) Map area in global overview state; each colored block is a conference activity; hovering reveals details of the activity; clicking on it zooms in (both in X and in Y) to show more detail. (b) Region for saving a user’s favorite sessions, highlighted scheduling conflicts. (c) The X-axis is a zoomable timeline; first click zooms to expand the corresponding day, second click zooms back out to the original view. (d) Color-coded session category labels; hovering will highlight (via embossing) all the sessions in the corresponding category. (e) Reset view button, for restoring the overview state. (f) The search box; search results, updated on each keystroke, show up on a panel on the left, and corresponding sessions are highlighted on the map. (g) Collaboration corner widget; hovering displays comments associated with the corresponding session, click pops up a commenting/annotating widget. (h) Switches for controlling the current view mode and displayed time zone.

timeline. *Tabular Timeline* differs from existing zoomable timeline visualizations such as [4, 19, 24] in that it uses a tabular layout to maintain the “look-and-feel” of a calendar rather a scatter plot. At the same time, regardless of the zooming level and the state of the system, time information is always represented in the horizontal axis in a timeline fashion. This behavior differs from existing calendar interfaces such as MS Outlook, Google Calendar or DateLens [3], where the time representation usually jumps between axes depending on the viewing mode (day, week, month etc.). Both informal feedback and our user study show that maintaining a consistent timeline in a zoomable interface helps maintain a consistent user experience.

Along the vertical axis of *Tabular Timeline* are tracks which can be independently expanded and collapsed via mouse clicks (similar to the design of nodes in DOI trees [7]). The colored rectangles within tracks correspond to sessions. When a track is collapsed, concurrent sessions are collapsed into a single rectangle. Clicking on a session in an expanded track will bring up its details pop-up (Figure 2).

Clicking on the “Related Information” button on the details pop-up creates a large frame populated with an external web page pertinent to the session. This frame can be closed to return to the previous visual context, or detached to a separate browser tab.

In addition to discrete zooming (by clicking on different regions of the view), the *Tabular Timeline* supports continuous zoom and pan via mouse dragging. Hovering over a category label on the top right region highlights corresponding sessions on the *Tabular Timeline*. With this quick action the user can see how sessions of a certain type are distributed throughout the conference.

Regardless of the current zoom level, clicking the “reset view” button (Figure 1.e) on the top left will reset the *Tabular Timeline* to its original global overview state.

Embedded Search

When the user types into the search box, search results, updated on each keystroke, show up on a panel on the left, and corresponding sessions are highlighted on the map

(Figure 3). For example, typing in "Washington" and restricting the search to titles and names will quickly show the sessions for papers authored by individuals from the University of Washington. Clicking on a search result shows/hides its session details popup in place on the map.

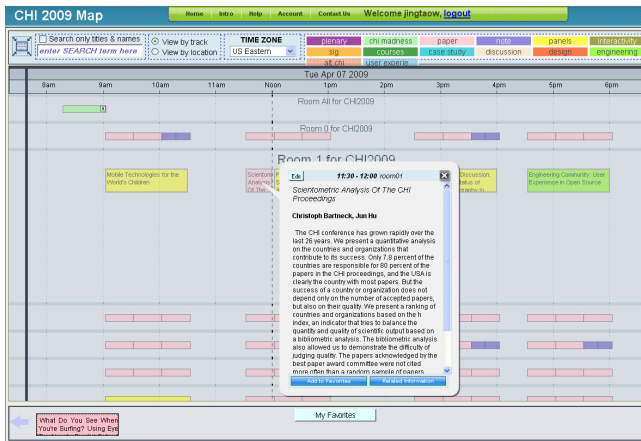


Figure 2. The *Tabular Timeline* zoomed into the day view, with one track expanded and the details pop-up open.

Active Corners

Rather than allocating designated regions to support personalization and asynchronous collaboration, we implanted a compact decorator widget named *Active Corners*. Active Corners serve both as an awareness indicator and a trigger for supported features. In the current implementation, the top-left corner of an event rectangle is used as the “*personalization corner*” and the top-right corner is used as the “*collaboration corner*”. In the future, the two corners on the bottom of an event rectangle can also be mapped, e.g., for showing visiting statistics. Each Active Corner can have three states – 1) the default state is the awareness state, shown as a small icon at the corresponding corner to indicate Boolean information such as whether this session is marked as a favorite by the current user, or whether there are comments associated with the session. 2) If a session is expanded, hovering over an Active Corner can provide a quick preview of information associated with that corner. 3) If a session is expanded, clicking on the corresponding *Active Corner* can trigger an action, such as modifying the favorite state of the session or popping up a widget for browsing and adding comments on the session. We describe in detail the two *Active Corners* currently in use in Event Maps below.

Personalization/Favorites Management

After logging in, a user can add an event to “My Favorites” by simply checking the check box at the “*personalization corner*” of a session or clicking the “Add to Favorites” button in its details pop-up. Proxies for favorite session are placed at the bottom of the screen. A user can remove a favorite by unchecking the check box the same button in the details pop-up, now labeled “remove from favorites”. Any sessions that a user has added to his favorites are shown

with a check mark on the “*personalization corner*” thus giving the user a quick indication of when she is free to schedule other activities.

Asynchronous Collaboration

In contrast to collaborative visualization web sites such as sense.us [12] and swivel [23], which were designed to support and promote collaborative sense-making, we believe that different Event Maps users may have different goals in mind. Some might only be interested in getting the schedule information they need quickly and exhaustively, while others might be interested in exchanging information or discovering interesting things from other participants. Thus, our major design principle for the asynchronous collaboration feature is that it can be ignored easily by people who do not need it, yet be discovered and accessed conveniently by interested parties.

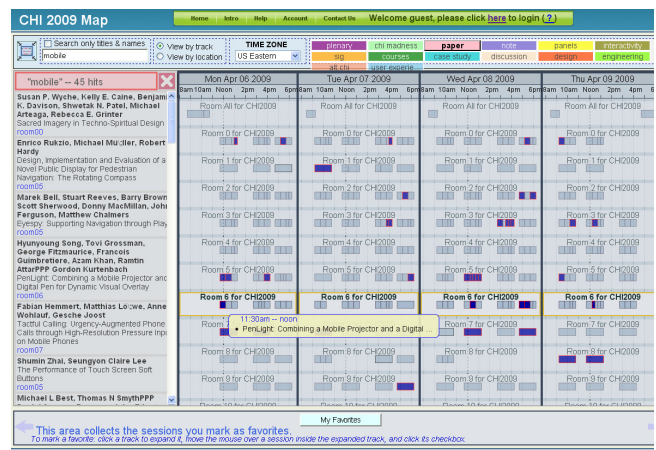


Figure 3. The search feature of Event Maps.

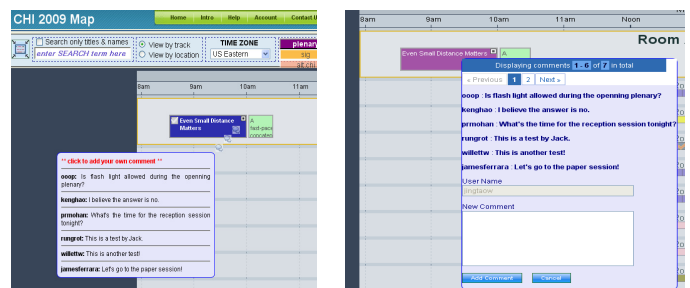


Figure 4. Collaboration features in Event Maps. (Left: a solid square on the top right corner indicates that there are associated comments; hovering over the indicator will display them. Right: clicking on the indicator brings up a comment-entering widget)

Asynchronous collaboration features are accessed from the top-right *Active Corner* (a.k.a. “*collaboration corner*”) of any session. Mouse hovering on it will display comments associated with the session (Figure 4, left). After logging in, mouse clicking on the *collaboration corner* will pop up a commenting/annotating widget (Figure 4, right). A tiny black dot will show up on the “*collaboration corner*” of sessions with comments or user generated tags. Again,

adhering to our design principle, the user does not have to go to a new destination to access the collaboration features.

Like in a wiki system, a user who has logged in, optionally with administrator privileges, can edit the event details in place by clicking an “edit” button on event details pop-up.

THE EVOLUTION OF EVENT MAPS

Event Maps draws its design from its earlier prototypes and a broad set of inspiring earlier works. While the individual features of Event Maps represent variations of existing approaches, we have integrated several techniques to create a novel application that allows users to explore and share information about multi-track conferences in a way that was not possible to date. As of this writing, Event Maps has gone through three major rounds of iterative design. Here we report the insights and lessons learned during this process.

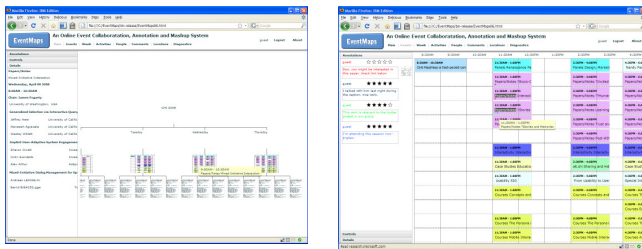


Figure 5. The first iteration of Event Maps implemented in Adobe Flex. (Left: In overview, events are organized in a tree structure. Right: The zoomed in view of a day)

Our first design (P1) used a hierarchical, tree style zoomable layout as the primary interface as shown in Figure 5 and was drastically different from the traditional look and feel of a calendar. P1 was implemented as a Rich Internet Application in Adobe Flex 3.0. The primary view of P1 is a zoomable “event tree”. Conference events are aggregated and placed on different levels based on properties, such as day, time, topic, speakers, location and keywords of the event by using an approach similar to Potter’s Wheel [20]. We also implemented features such as semantic zooming [2], brushing, and drilling down [6] in P1. To support collaboration we implemented a double-linked commenting feature. Comments are shown in a foldable panel on the left, a separate comments view and as miniaturized, color encoded dots on conference events.

Our first prototype received mixed feedback from early testers – on the one hand, many people were excited by the rich interactions provided by our prototype. On the other hand, we also identified significant usability problems, especially from people who were less technically savvy.

For example, the tree-based interface requires extra learning time before people can take advantage of the rich interactions supported by the interface. Based on the authors’ rough estimation, it took a novice user around ten minutes to get familiar. A frequent question we received was “*how are conference events placed at different levels?*”

In retrospect, a tree-based hierarchy might not be the most intuitive conceptual model for conference events. Promoting a drastically new interface should take into account leveraging users’ expertise in existing interfaces.

In the process of creating P1, we also discovered the effectiveness of certain features, which we carried out to future prototypes. Two important lessons were:

1. It is effective to use semantic zooming [2], i.e., revealing information differently based on the zooming level, to keep a good balance of legibility vs. visibility in a zoomable interface for temporal data.
2. Although the entire schedule information might easily overwhelm the human eye, it can be stored quite compactly on a computer in standard format. For example, the entire CHI 2009 conference schedule, even with paper abstracts not available on the official website, is around 498KB in plain text. With compression this goes down to 150KB. We learned that we can provide a more responsive system by sending the whole schedule at the start rather downloading session details on demand via fine-grained web services.

We built the second system (P2) based on the lessons learned from the first prototype. We switched the browser side technology from Adobe Flex to JavaScript/AJAX to enable access from even mobile device browsers. The look and feel of the second prototype is similar to the current prototype (P3) shown in Figure 1. The biggest change we made in P2 is the introduction of the *Tabular Timeline* interface. P2 was deployed in an intra-company conference named SRS 2009 (described in detail in the next section). Based on both quantitative analysis of access logs on the server and qualitative interview results with 6 active users of Event Maps during the conference, we built P3, and ran a 12 participant user study to compare P3 directly with a state-of-the-art conference schedule website (that of CHI 2009 [8]). P3 will be hosting an upcoming public conference - IEEE SCC 2009², International Conference on Services Computing between September 21 and September 25, 2009. The major changes made between P2 and P3 were:

- Addressed usability problems identified in previous studies. Two examples, the search box and the reset view button, will be detailed in the next section.
- Designed and implemented the *Active Corners* feature for personalization and asynchronous collaboration
- Built the server side administrative interface and infrastructure for creating/importing new conferences and updating existing events efficiently.
- Improved the backend logging, analytic functions and interfaces for conference organizers.

² Event Maps for SCC 2009 can be accessed from the conference homepage: <http://conferences.computer.org/scc/2009/index.html>

With an improved server side administrative interface for managing conference events, it becomes convenient for administrators to import and maintain the schedule. We have tested P3 with event data from 19 conferences so far, some sample conferences we tested are listed in Table 1.

Conference	Total Events	Total Categories of Events	Duration (days)	Total Number of Rooms
CHI 2009	475	14	4	14
CHI 2008	467	12	4	14
Lotusphere 2009	351	7	5	19
SCC 2009	241	14	5	7
OSCON 2009	306	24	5	19
SRS 2009	126	8	4	9
RailsConf 2009	115	4	4	10
MySQL 2009	190	22	4	16

Table 1. Statistics of some sample conferences supported by Event Maps

IMPLEMENTATION

The current implementation of Event Maps includes two parts: a client side and a server side. The client side is implemented in JavaScript/AJAX and can run on major web browsers such as FireFox, Safari, Internet Explorer, and Chrome. The server side is implemented in Ruby on Rails 2.3. IBM DB2 is used on the server side for data persistence. Excluding third party libraries and code for unit testing, the current Event Maps implementation includes a total of 11032 lines of code in JavaScript that runs in the browser and a total of 12980 lines of code in Ruby.

The JavaScript client is object-oriented, with objects corresponding to the major model, view and controller elements. It uses only standard JavaScript with no special libraries, and relies heavily on dynamic manipulation of the DOM and associated style sheets. The various animations are performed via timeout loops, and the DOM is only partially recomputed during an animation for improved performance. As browser support for JavaScript efficiency improves (the recent upgrades to Firefox 3.5 and IE 8 are good examples), so does the performance of Event Maps.

In our system, the client side and the server side use JSON over HTTP to exchange user-profile data and user-generated data. The conference schedule data (including detailed times and descriptions) are sent to the client as a single text file (around 498KB for CHI 2009 before compression) in iCalendar format [21]. By adopting open standards, when necessary, both the client side and the server side can be changed to a different technology in the future without impacting the interoperability.

As a result of our decision to use only standard JavaScript and AJAX, Event Maps can run on a wide range of operating systems and browsers. For example, Figure 6 shows two screen shots of Event Maps running on an

iPhone. Event Maps maintains a similar user experience to that on desktop computers and portable, and the pinch operation on the iPhone maps rather nicely to the basic zoom operation. That said, the effectiveness of Event Maps on small form-factor devices needs to be evaluated. We plan to study this issue, as well as providing some user interface optimizations for portable devices e.g., leveraging multi-touch gestures as alternatives to mouse-hovering.



Figure 6. Screen shots of Event Maps running on iPhone (left: CHI 2009 schedule in overview mode, right: zoomed in view with in-place event editor enabled)

USER STUDY

We now report both the insights and lessons learned during the iterative design process for building Event Maps.

Informal Early User Study and Feedback

Initial user feedback on an early version of P2 was obtained via an informal study during an hour-long department meeting. After a short general introduction to Event Maps with no training, the 16 participants were given a link to a questionnaire hosted on SurveyMonkey.com that guided them through 3 tasks as if they were conference attendees (before, during, after the conference). Each task required some sort of result (e.g., a session name) and a free-form response on how well Event Maps supports the task scenario. The final page listed 18 Event Maps features and asked, for each one, to check “didn’t use it”, “don’t like it”, “it’s OK”, or “nice!”. Some noteworthy lessons we learned from this study were:

Support unexpected interactions: double-click caused a bug in a place in the code we had designed for single-click.

Don’t be overly cute: in the “related information” frame we used a semi-transparent red X in the middle. Some users did not see it, and were stuck, unable to close the frame!

Support freedom of use: the following comment led us to make the “related information” URL explicit: “I found the recording of this talk, which was very valuable, but I wish the page would open in a new window, so that I can add it to my Firefox bookmarks for listening it later.”

Nudge them (within reason): Paradoxically, it was a good idea to make the free-form questions mandatory – that’s where we learned the most. It would have been good to also ask users who did not like a feature to explain why.

Live Deployment

We conducted a live deployment of Event Maps version P2 for an internal Software Research Strategy meeting (SRS 2009) held across eight IBM Research sites worldwide. It had 126 events spread over four days (Table 1). Participants could either attend local events or watch live online broadcasts of remote sessions. A link for Event Maps was posted on the primary conference website and an announcement was made during the opening plenary.

During the course of this conference, we collected a total of 3297 significant logs³ from 298 unique IP addresses. Figure 7 reveals the visitation frequency from these unique addresses. Similar to most web-based activities; the visitation frequency conforms to the power law distribution.

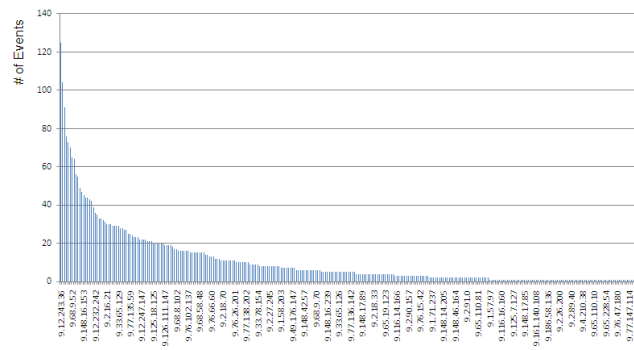


Figure 7. Visiting frequency by IP during SRS 2009. No IP includes events from two or more registered users.

76.1% of users were Windows users and 23.9% of users were Mac users. Firefox 3 was the most popular browser (52.9%) and Internet Explorer 7 accounted for 23.2% percent of the total access. 13.1% used Safari, and 6.4% used Chrome (figure 8). The effort we put to target and test on multiple browsers had paid dividends.

While nearly 69% percent of the traffic was from three sites in U.S., there were visitors who came from ten different time zones of the globe. This fact confirmed the importance of the “time zone” selection feature we implemented.

By analyzing the traffic log, we also identified several areas for improvement in Event Maps:

- Despite the relatively large number of visitors, only 19 visitors (around 6%) registered an account on Event Maps, most people used the web site anonymously. As

³ Access logs generated by team members of the Event Maps project were excluded from analysis.

a result, they didn’t have a chance to try features such as persistent favorites or preferred time zone. Only 53 users re-visited Event Maps after more than one hour.

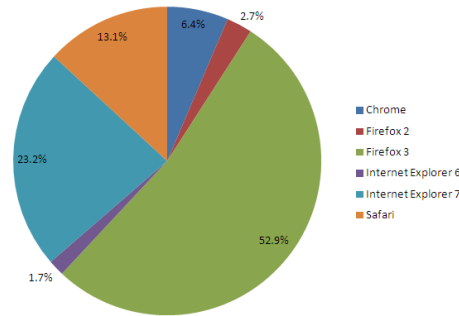


Figure 8. Distribution of browsers for SRS 2009 deployment.

- To our surprise, the search function was rarely used during the deployment; most interactions were drill down operations for obtaining detailed information.
- Although we provided a simple tutorial in a highly visible tab on the website, less than 3% of the users actually visited it. Literally no one used the feedback form on the website during the study though we did receive suggestions from several via e-mail.

Interview Results

After the conference, we identified frequent users that had registered, and sent them interview invitations. 6 users (2 females and 4 males) accepted the invitation and agreed to have an interview with us regarding their experiences with Event Maps (four interviews were conducted face-to-face; two were conducted over the phone). No incentives were offered to participate. The interviews were semi-structured including questions on how they heard of Event Maps, their experiences with specific features, such as favorites and search, and their opinions on prospective features to be implemented. We also went through open questions such as likes, dislikes, suggestions and comments. The Interviews lasted 30 to 40 minutes; five were audio-recorded with explicit consent of the interviewees.

Highlights of the face to face interviews:

Tabular Timeline is Intuitive: most interviewees said that the zoomable interface of Event Maps was easy to master and use. The color-coded events and the hovering-to-highlight category labels were used by most interviewees. In general, they all liked this feature and indicated that it improved their awareness of ongoing conference events.

Not looking at any help, mouse click as the probe: Rather than looking for a help page, almost all the users explored the interface directly. Clicking and hovering the mouse on various regions of the screen were the two most popular probing activities. At the same time, dragging, drag and drop, right button clicking and keyboard shortcuts (one

interviewee “didn't even think of it!”) were rarely used to explore the interface. As a lesson learned, when we add new features to Event Maps, we always make sure that they can be discovered by left mouse clicking or hovering.

Interests in other people's opinions: In addition to all the “official” information (including slides and video), most interviewees indicated their keen interest in knowing other people's opinions and interests on conference events, especially people they know and trust.

As detailed in the design evolution section, most of the new features we added in P3 were driven by the lessons and insights from the SRS 2009 live deployment. In addition to new features, we also made various adjustments to the UI to address usability problems we had identified.

Sample UI Revisions

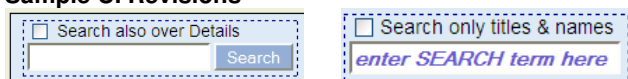


Figure 9. Sample UI change after the SRS 2009 deployment

Figure 9 shows a sample UI revision we made – since the keyword search function is completely locally without internet connection, Removing the search button suggests that searching is responsive and encourages the users explore with different keywords. We also noticed via analyzing the access logs that the most common action after accessing the detailed information of a session was going back to the global view, so we added an explicit the reset view button (figure 1.e) in addition to keyboard shortcut on the top left corner to support this action. The reset view button now becomes one of the most frequently access features in Event Maps.

Lab Study

After completing P3, we ran a formal lab study to evaluate the effectiveness of Event Maps in completing common tasks in a conference, especially its relative performance when compared with an existing state-of-the-art conference web site for accessing the same schedule information.

Participants

12 people (3 female, 9 male) between 22 – 36 years of age (mean = 26.1) participated in the user study. Subjects were collected via recruiting emails to multiple internal mailing lists in a university. All participants were right-handed. All participants have some previous experiences in attending a conference before the study. 10 subjects had an educational background in engineering, one in business administration and one in linguistics.

Apparatus

The experiment was conducted using a Lenovo Thinkpad T400 Laptop computer with an external Microsoft Laser Mouse 6000 as the pointing device. The computer had an Intel Core 2 Duo 2.26GHZ CPU, a 14 inch LCD monitor at a resolution of 1440 by 900 pixels. The operating system

was Windows XP SP3. Firefox 3.5 was used as the browser for all the tasks in our study. The Event Maps server used in this study was hosted on a Dell PowerEdge server running Ubuntu Linux 8.04 server edition, with a Pentium 4 2.8 GHZ CPU and 2GB Ram.

We pre-loaded 14 conferences to the Event Maps server, including those listed in Table 1. Subjects could freely browse these conferences on Event Maps during the exploration stage of the test. Subjects were asked to complete tasks for ACM CHI 2009. We also mirrored the ACM CHI 2009 conference schedule pages [8] on the same server as an alternative technology. The two major reasons for mirroring were to have both systems studied have the same network access speed, and to enable logging of users' visitation behavior on the web pages during the study. No modifications were made for to mirrored web pages.

Experimental Design

The experiment used a within-subject design, using two sets of comparable tasks. Each set included 12 tasks. The tasks were created by interviewing frequent conference participants about their high-frequency activities in a conference and by consulting domain experts in CSCW.

The nature of the tasks in each set is summarized below – 1) Locating the starting time of a talk; 2) Locating an author based on a keyword in the title of a talk; 3) Planning a short-term (2 hour) schedule for a given interest; 4) Determining the time and location of a talk by a given author on a given day; 5) Obtaining details of an event, given the time and event category; 6) Planning a long-term (half day) schedule for a given interest; 7) Counting events in a given category within a certain time span; 8) Obtaining details of an event given relatively vague time constraints; 9) Obtaining details of an event given accurate time and location information; 10) Finding a co-author of a given author; 11) Comparing the popularity of two technologies (in terms of keyword frequency); 12) Counting events in a given category within part of a day.

Brief tutorials (around 7 minutes) were provided prior to each set of tasks on each system. During the tutorial, the experimenter first demonstrated major features of the corresponding system. Then we encouraged users to spend some time to explore the system. We waited to start the actual tasks until the participant indicated explicitly to the experimenter that he/she was comfortable with the current interface. Most participants spent around 5 – 10 minutes in exploring each interface before starting the actual tasks. The participants performed the set of 12 tasks mentioned above using each calendar. The order of calendar use and task set for the calendar were both counterbalanced in order to minimize the effects of training, or the possibility of one task-set being slightly more difficult than the other. However, tasks within a set were not randomized.

Results

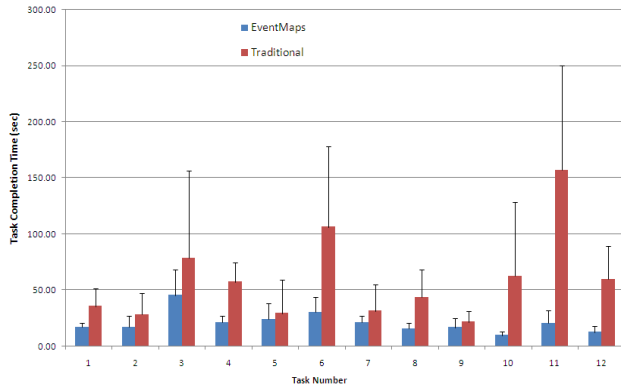


Figure 10. The average completion time (seconds) vs. task number. The error bar indicates the standard deviation of corresponding completion time

The average completion time of all the tasks are shown in Figure 10. Average completion times for Event Maps were shorter than on the official website in *all* tasks. Using Event Maps, the average completion time for each task was less than 46 seconds; while in the traditional web site, the average time for some tasks, e.g. task 6 (long planning) and task 11 (counting), could be more than 2 minutes. However, the differences are statistically significant only for tasks 1, 4, 6, 8, 10 and 12 (task 1: $F(1, 11) = 3.88, p < 0.005$), task 4 ($F(1, 11) = 6.93, p < 0.001$), task 6 ($F(1, 11) = 4.00, p < 0.005$), task 8 ($F(1, 11) = 4.39, p < 0.005$), task 10 ($F(1, 11) = 2.76, p < 0.02$), task 11 ($F(1, 11) = 4.86, p < 0.001$), task 12 ($F(1, 11) = 5.82, p < 0.001$).

The difference in average completion time for task 2 ($F(1, 11) = 1.67, p = 0.12$), 3 ($F(1, 11) = 1.54, p = 0.15$), 5 ($F(1, 11) = 0.92, p = 3.76$), 7 ($F(1, 11) = 1.34, p = 0.21$), 9 ($F(1, 11) = 1.68, p = 0.12$) were not statistically significant.

An interesting observation was the performance of task 3 (planning a 2 hour schedule for a given interest) and task 6 (planning a similar half-day schedule) in both conditions. Unsurprisingly, on the traditional web site, the time spent in completing the task increased as the required planning time-span increased, as more page switching was needed. However, although the difference is not statistically significant, on average it took less time to schedule a longer trip in the Event Maps condition! We attribute this observation to two reasons: First, these two tasks can be completed in the global view mode with either mouse hovering or keyword search, as there is no view switching involved despite the extended planning time-span. Second, there might be learning effects between task 3 and task 6, since the order of tasks in each test set is not randomized.

Figure 11 shows the average “perceived” difficulty for each task on a 5-point Likert scale (5 means extremely easy, 3 means neutral and 1 means extremely difficult). In accordance with the actual performance, Event Maps was perceived easier to use than the traditional web site in most

tasks. Interestingly, task 5 (Locating the name of a panel that meets the provided date and time constraints) was perceived to be easier to finish via the traditional web interface even though the average completion time of that task was slightly shorter using Event Maps.

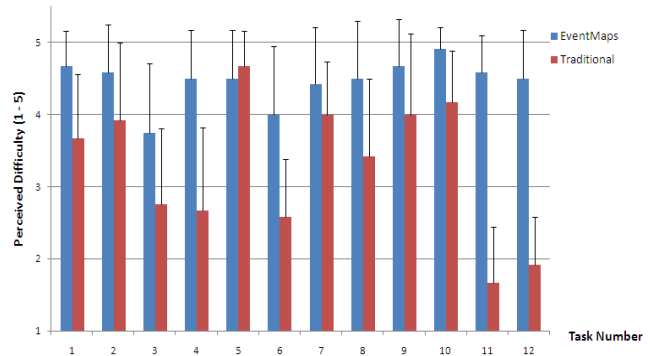


Figure 11. The average perceived difficulty for each task. (1 = extremely difficult, 3 = neutral, 5 = extremely difficult). The error bar is the corresponding standard deviation.

In the closing questionnaire, many users indicated explicitly that they like the zoomable *Tabular Timeline* layout and the search feature in Event Maps. Sample quotes include “[Event Maps] links time with location clearly.” “Presentation is clear, Integrated search is very helpful and is more helpful even than the type ahead search in the [Firefox] browser which I use constantly and leveraged to complete the tasks on the old web site.” “The search ability, and the representation of all events in categories in the timeline fashion made [it] easy for navigation”

During the study, at least 4 users clicked on category labels while exploring the system. When asked why, the given answers were similar: those labels are button-shaped, when the mouse is over a label, that label gets embossed and looks exactly like a button, implying that they are clickable. Some users noted that when hovering on the “course” category label, they could not see anything because those courses are out of the current view port - they suggested that it might be better to pan the current view automatically to make most of the “course” events visible. One user suggested that when a category is clicked, events of all other categories become homogeneously dark, until the same category label or other category labels are clicked. In this way, the users could scroll the current view up and down to get some global feelings about a specific category. Interestingly, the iPhone already supports “sticky” category highlighting in lieu of hovering support!

Satisfaction question regarding Event Maps	Mean	Stdev
Overall, I am satisfied with this system	4.58	0.67
It was easy to learn	4.25	0.62
Improved my awareness of ongoing events	4.50	0.67

Improved my ability to find info on a specific topic	4.75	0.45
Helped me to keep track of events I'm interested in	4.25	0.45
Saved my time in locating the information I need	4.83	0.39
I could effectively complete the tasks in my mind	4.42	0.67
I was able to complete the tasks and scenarios quickly	4.42	0.51
The "look-and-feel" of this system was pleasant	3.92	0.67
Organization of information in this system was clear	4.42	0.51

Table 2. Summary of the satisfaction rating in the closing questionnaire (on 5 point Likert scale, 1 means strongly disagree, 3 means neutral, 5 means strongly agree)

Table 2 summarizes the average satisfaction rating after using Event Maps. Overall, the ratings were highly favorable. Participants strongly believed that using Event Maps could save their time in locating the information they need (mean = 4.83, stdev=0.39). The "look-and-feel," especially the color theme of the current system was considered to still have significant room for improvements.

DISCUSSIONS AND FUTURE DIRECTIONS

Based on evidence from studies and deployments we have demonstrated that users feel Event Maps provides a good mental map and faster access to several typical pieces of information conference attendees need through its use of a zoomable interface and by exposing information on demand conveniently via tool tips and mouse actions.

Our practical experience in building Event Maps leads us to the following observations:

- "It is faster to make a four-inch mirror, then a six-inch mirror than to make a six-inch mirror" - The *Thomson's Rule for First-Time Telescope Makers* also applies to building interactive systems.
- Build simple components as services with clean interfaces for adoption in broader contexts.
- Avoid premature optimization to get to the right design faster.
- Disk is cheap, so log and analyze more data than less.
- Scripting languages speed up prototyping.

Some areas for future work include the following:

- Customize Event Maps for access from popular mobile devices with particular attention to usability, i.e., interaction in mouse-less environments, legibility on small screens, and limitations in compute power.
- Use information provided by attendees, including browsing patterns within Event Maps, to improve the conference experience.
- Enable micro-tagging on a wide variety of conference material – including sections of papers, authors, institutions, recordings, etc.

- Make the conference site the hub for activity even after the end of the conference to benefit both attendees and those not fortunate enough to get the live experience.
- Incorporate indoor floor maps to enable several location-based services pertinent to conference attendees.

CONCLUSION

Event Maps is a highly interactive web-based collaborative calendaring system optimized to provide quick access to information at multiple levels of detail for large, multi-track conferences. Further, Event Maps can be used as a "collaboration space" for visualizing and collaborating on large, complex temporal data with a hierarchical nature.

The Event Maps system has been refined twice based on quantitative and qualitative feedback from user studies, lab studies and deployments. The feedback we have received thus far on how Event Maps enhances the conference attendee's user experience has been very encouraging and we will thus be pursuing even larger deployment opportunities to gain insights into how conference collaboration substrates get used during various stages of a conference and from different device form factors, build a prioritization of the feature set based on several factors, and use the findings to allow Event Maps to meet the user's needs in various times and situations.

We will analyze and report in depth the insights from face-to-face interviews, lab based studies and real world deployments throughout the iterative design process. In addition to a holistic evaluation of the overall effectiveness of Event Maps, we are also interested in getting a deeper understanding of the impact of each specific feature in its corresponding design space.

ACKNOWLEDGEMENTS

We thank Dan Coffman, Alpna Tiwari, Jon Munson, Tom Erickson, Sam Adams, Dan Gruen, Mercan Topkara and Jeffery Nichols for their help and constructive suggestions.

REFERENCES

1. Ahlberg, C, Shneiderman, B., Visual Information Seeking Using the FilmFinder, In Proc. ACM CHI 1994
2. Bederson, B., Fisheye Menus, In Proc. UIST 2000
3. Bederson, B., Clamage, A., et al., DateLens: A Fisheye Calendar Interface for PDAs, In Proc ACM ToCHI 2004
4. British History Interactive Timeline <http://www.bbc.co.uk/history/interactive/timelines/british/>
5. Brzozowski, M., Carattini, K. et al., groupTime: preference based group scheduling, in Proc CHI 2006
6. Card, S., Information Visualization, In the Human Computer Interaction Handbook, pp 509-543, Lawrence Erlbaum Associates, 2008

7. Card, S., and Nation, D. Degree-of-interest trees: a component of an attention-reactive user interface. In Proc. AVI 2002
8. CHI 2009 Conference Program <http://www.chi2009.org/Attending/AdvanceProgram/monday.html>
9. Dasu, T., Johnson, T., et al, Mining database structure; or, how to build a data quality browser, In Proc. SIGMOD 2002.
10. Furnas, G., The Fisheye Calendar System. (Report No. TM-ARH-020558). Bellcore, Morristown, NJ. 1991.
11. Heer, J., Agrawala, M., Design Considerations for Collaborative Visual Analytics, IEEE VAST 2007.
12. Heer, J., Viégas, F., Wattenberg, M., Voyagers and Voyeurs: Supporting Asynchronous Collaborative Information Visualization, In Proc. ACM CHI 2007
13. Hochheiser, H., Shneiderman, B., Dynamic Query Tools for Time Series Data Sets, Timebox Widgets for Interactive Exploration, Information Visualization 3, 1, March 2004
14. Jensen, C., Snodgrass, R., Temporal Data Management, IEEE Trans. Knowledge and Data Eng., vol. 11, pp. 36-44, 1999.
15. Keogh, E., Chakrabarti, K., Pazzani, M. and Mehrotra, S., Locally adaptive dimensionality reduction for indexing large time series databases, In Proc. ACM SIGMOD 2001
16. Mackinlay, J., Automating the Design of Graphical Presentations of Relational Information, pp. 110-141, ACM Trans. on Graphics, Vol. 5, No. 2, April 1986
17. Neustaedter, C., Brush, A., et al., The calendar is crucial: Coordination and awareness through the family calendar, In Proc Transactions on Computer-Human Interaction (TOCHI), April 2009.
18. Plaisant, C., Milash, B., et al, LifeLines: Visualizing Personal Histories, In Proc. ACM CHI 1996
19. Plurk, <http://www.plurk.com/>
20. Raman, V., Hellerstein, J., Potter's Wheel: An Interactive Data Cleaning System. VLDB 2001.
21. RFC 2445 Internet Calendaring and Scheduling Core Object Specification <http://tools.ietf.org/html/rfc2445>
22. Rogers, T., Conferences and Conventions, Second Edition: A global industry, Butterworth-Heinemann; 2 edition, 2008
23. Swivel. <http://www.swivel.com>
24. Timeline, Web Widgets for Visualizing Temporal Data <http://www.simile-widgets.org/timeline/>
25. Wang, T., Plaisant, C., et al, Aligning Temporal Data by Sentinel Events: Discovering Patterns in Electronic Health Records, In Proc ACM CHI 2008.
26. Wattenberg, M., Kriss, J., et al., Many Eyes: a Site for Visualization at Internet Scale. IEEE Transactions on Visualization and Computer Graphics, Vol. 13, No. 6. (c 2007).