

Technical Disclosure Commons

Defensive Publications Series

January 08, 2015

CALENDAR EVENT MANAGEMENT

Mårten Jönsson

Follow this and additional works at: http://www.tdcommons.org/dpubs_series

Recommended Citation

Jönsson, Mårten, "CALENDAR EVENT MANAGEMENT", Technical Disclosure Commons, (January 08, 2015)
http://www.tdcommons.org/dpubs_series/4



This work is licensed under a [Creative Commons Attribution 4.0 License](https://creativecommons.org/licenses/by/4.0/).

This Article is brought to you for free and open access by Technical Disclosure Commons. It has been accepted for inclusion in Defensive Publications Series by an authorized administrator of Technical Disclosure Commons.

CALENDAR EVENT MANAGEMENT

ABSTRACT

A calendar event management system provides hassle free management of calendar events. The system receives a calendar invite for an event at a specified time slot for an invitee. On receiving the calendar invite, the system identifies an existing event scheduled at the specified time slot in a calendar for the invitee. Then, the system determines that the event has higher priority than the existing event. Further, on determining that the event has higher priority than the existing event, the system adds the event to the invitee's calendar.

PROBLEM STATEMENT

An inviter can add participants to a calendar invite in a calendar application to invite them to a meeting. The meeting can be a conference call, a video call, or a meeting that requires physical presence of the participants. When a large number of participants are required to attend the meeting, it becomes difficult for the inviter to go through the calendars of all the desired participants and find a time slot at which each of the participants is available. Moreover, due to busy schedules, scarcity of meeting rooms, and minimal overlapping time between time zones, it is challenging to find a time that is convenient for a large group of people. Thus, scheduling event invites can be administratively heavy and cumbersome, and can still result in overlapping meeting time slots for the participants. A more efficient system to handle calendar invites is described.

CALENDAR EVENT MANAGEMENT SYSTEM

The systems and techniques described in this disclosure relate to a calendar event management system for managing calendar invites for events. The system can be implemented for use in an Internet, an intranet, or another client and server environment. The system can be implemented locally on a client device or implemented across a client device and server environment. The client device can be any electronic device such as a mobile device, a smartphone, a tablet, a handheld electronic device, a wearable device etc.

Fig. 1 illustrates an example method 100 for managing calendar invites for events. The method can be performed by a system that manages calendar invites, for example, the calendar event management system. The system receives a calendar invite for an event at a specified time slot (Block 102). The calendar invite can be sent from an inviter to an invitee. The event can be any activity that the inviter is inviting the invitee to attend, e.g., a meeting, presentation, party, and dinner. The calendar invite can include a description and a location of the event, and a time slot that it is to occur on the invitee's calendar. The system receiving the calendar invite can access an account associated with the invitee. For example, the system can schedule the event on the invitee's digital calendar at the specified time slot. Alternatively, or additionally, the calendar invite can also be sent as an email to the invitee's email account.

On receiving the calendar invite, the system identifies an existing event scheduled at the specified time slot on the invitee's calendar (Block 104). The existing event can be any calendar event that already exists on the invitee's calendar at the time the system receives the calendar invite. The existing event can be scheduled at a time that partially or fully overlaps with the

specified time slot. The existing event might have been added automatically to the calendar by the system upon receiving the request for this event. For example, the invitee may have earlier received a calendar invite for another event for the same time slot. The invitee may or may not have accepted this event. Alternatively, the existing event may have been added manually by the invitee.

The system further determines that the event for which the calendar invite is received has higher priority than the existing event (Block 106). The system can determine the relative priority of the two events based on various factors, e.g., the respective inviters of the two events, the invitees for the two events, and the urgency of the event.

The system can determine that the event for which the calendar invite is received has higher priority than the already existing event based on the respective inviters of the two events. For example, higher priority is given to the event sent by the inviter with greater seniority or higher in a chain of command. The system checks the invitee's organizational, e.g., place of work, chain or structure to identify which inviter is more senior or higher in command. For example, the system identifies that the existing event is from the invitee's peer and the event for the received calendar invite is from the invitee's manager. Because invitee's manager is higher in command than invitee's peer, the system determines that the event has higher priority than the existing event.

Additionally, or alternatively, the system can determine that the event for which the calendar invite is received has higher priority than the existing event based on the respective invitees for the two events. The system analyzes the respective groups of people that have also been invited to the two events and determines which group of people the invitee more frequently

meets with. The system determines a higher priority for the event with other invitees that the invitee more frequently meets with. For example, the system can analyze all events that exist on the invitee's calendar and identify the frequency that the invitee meets with different people. This frequency can then be applied to the people invited to the two events to identify the event with the group of people that the invitee meets with more frequently. Additionally, or alternatively, the system can determine that the event for which the calendar invite is received has higher priority than the existing event based on the respective urgency of the two events. The system determines a higher priority to the event that is about something urgent. The system can extract keywords from both the calendar invite for the event and the existing event. The extracted keywords are compared with keywords in the invitee's email. The system identifies keywords from recent emails in the invitee's email that match the extracted keywords from the currently received event. Based on the degree of matching keywords, the system determines that the event for which the calendar invite is received is more important or has higher priority than the existing event.

Additionally, or alternatively, the system can determine that the event for which the calendar invite is received has higher priority than the existing event based on a combination of the respective inviters, invitees, and urgency of the two events. For example, the system determines a score for each of the characteristics and combines the scores for each of the respective events. The event with the greatest combined score is given higher priority by the system.

Further, on determining that the event has higher priority than the existing event, the system adds the received event to the invitee's calendar (Block 108). The system can add the event to the calendar and reschedule the existing event for another time. The system can access calendars of attendees of the existing event to check if the attendees have some available time slots which are close or near to the specified time slot. The system can select a new proposed time slot from one or more available time slots and send the attendees a new invite for the existing event at the new proposed time slot. Optionally, the system may also provide a reason for rescheduling the existing event to the attendees. Alternatively, the system can negotiate with other systems associated with the attendees to pick a new suitable time slot for the existing event. For example, the system can request the other systems associated with the respective attendees of the existing event to find a new time slot that is suitable for the attendees of the existing event. The other systems can check calendars of their respective users and communicate with each other to finalize a time slot that is non-overlapping and convenient for all. Alternatively or additionally, there may be situations when for rescheduling an existing event, other one or more existing events may also need to be rescheduled. The system may negotiate with systems of attendees of other existing events to find new time slots for rescheduling the other existing events. The systems can notify all the attendees about the new picked suitable time slot.

The subject matter described in this disclosure can be implemented in software and/or hardware (for example, computers, circuits, or processors). The subject matter can be implemented on a single device or across multiple devices (for example, a client device and a server device). Devices implementing the subject matter can be connected through a wired and/or wireless network. Such devices can receive inputs from a user (for example, from a

mouse, keyboard, or touchscreen) and produce an output to a user (for example, through a display). Specific examples disclosed are provided for illustrative purposes and do not limit the scope of the disclosure.

DRAWINGS

100

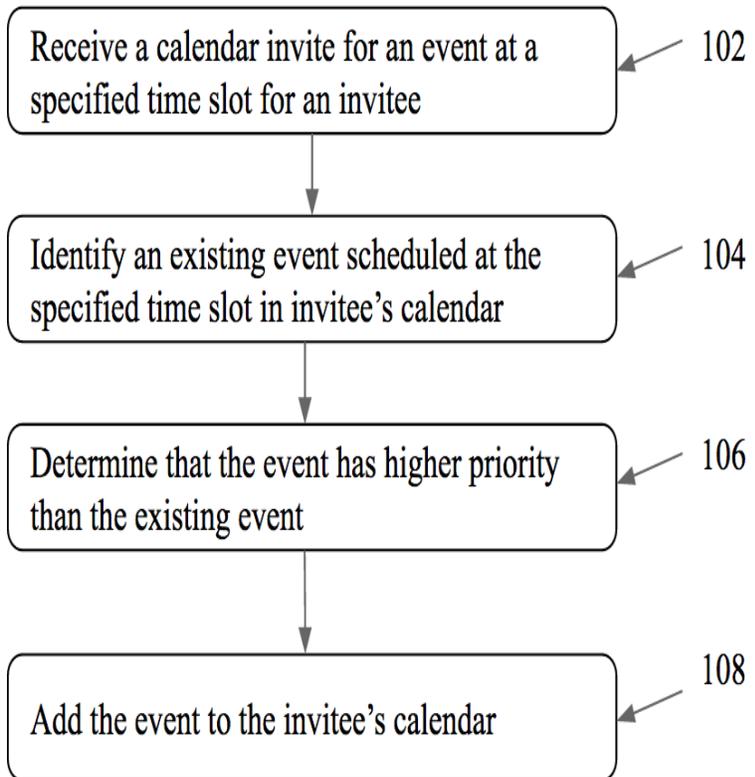


FIG. 1