ASSISTANT-TO-ASSISTANT CALENDARING
ASSISTANT-TO-ASSISTANT CALENDARING

ABSTRACT

A virtual, intelligent, or computational assistant (e.g., also referred to simply as an “assistant”) is described that is configured to work with other users’ assistants to schedule events (e.g., appointments, bookings, etc.) on behalf of a current user and the other user(s). For instance, user A may request that their assistant schedule an event with user B (e.g., user A may say “find and book a data that user B can join me for dinner”). User A’s assistant identifies the user referenced by the request (i.e., user B), communicates with user B’s assistant to negotiate a date/time for the event (e.g., a mutually convenient date and time based on both user A’s schedule and user B’s schedule), and creates an entry in user A’s calendar for the event at the negotiated date/time. The assistants may perform the date/time negotiation subject to authorizations received from their respective users.

DESCRIPTION

Assistants execute on counter-top devices, mobile phones, automobiles, and many other type of computing devices. Assistants output useful information, responds to users’ needs, or otherwise performs certain operations to help users complete real-world and/or virtual tasks. Some assistants may schedule calendar events involving a single user in response to requests from the single user. While scheduling calendar events involving a single user may be somewhat trivial, scheduling calendar events involving groups of people (i.e., calendar events requiring the coordination of a requesting user’s calendar with calendars of users other than the requesting user) may be more challenging.
The example system shown in FIG. 1 provides an assistant that facilitates multi-user event scheduling. For instance, when user B requests that the assistant schedule a meeting for user B and user A, user B’s assistant may communicate with user A’s assistant to identify a date/time for the meeting. Once a date/time for the meeting has been identified, the assistants may create entries in calendars of their respective users for the meeting. This way, other than providing authorization for their assistants to access and/or share calendar data for the date/time negotiation, user B and user A may schedule a meeting with minimal effort (i.e., may avoid the back and forth that occurs when trying to find a time for two people with crowded calendars).
The system of FIG. 1 includes one or more external systems and computing devices A–N communicating across a network with each of computing devices A–N executing an assistant.
that performs operations involving groups of people. The network of FIG. 1 represents a combination of any one or more public or private communication networks, for instance, television broadcast networks, cable or satellite networks, cellular networks, Wi-Fi networks, broadband networks, and/or other type of network for transmitting data (e.g., telecommunications and/or media data) between various computing devices, systems, and other communications and media equipment. Computing devices A–N represent any type of computing device, server, or other system that is configured to execute an assistant and communicate on a network. The external systems represent any type of server, mainframe, cloud computing system, or other computing system that is configured to support the assistants executing at computing devices A–N.

The external systems and computing devices A–N can be personal computing devices. In some examples, the external systems and/or computing devices A–N may be shared assets of multiple users. Examples of computing devices A–N include mobile phones, tablet computers, wearable computing devices, countertop computing devices, home automation computing devices, laptop computers, desktop computers, televisions, stereos, automobiles, and any and all other type of mobile and non-mobile computing device that is configured to execute an assistant. For example, computing device A may be a countertop assistant device and computing device N may be a mobile phone or automobile infotainment system.

An assistant executes across any combination of external systems one or more of computing devices A–N to provide assistant services to users of computing devices A–N. Examples of assistant services include: setting up reminders, creating calendar entries, booking travel, online ordering, sending messages or other communications, controlling televisions, lights, thermostats, appliances, or other computing devices, providing navigational instructions, or any other conceivable task or operation that may be performed by an assistant.
The assistant may be a single assistant with separate instances for each user. For instance, the assistant may have a first instance associated with user A and a second instance associated with user B. The instances may be a “copy” of the assistant or may be a set of information of the associated used. In some examples, each user may have a separate assistant.

As a user interacts with the assistant, the assistant may obtain personal information about the user. Examples of personal information include: habits, routines, preferences, notes, lists, contacts, communications, interests, location histories, and other types of user information. After receiving explicit permission from the user, the assistant may store, the personal information at user information data stores and in the course of providing assistant services, make use of the personal information stored at the user information data stores.

The external systems and computing devices A–N and the assistant treat the information stored at the information stores so that the information is protected, encrypted, or otherwise not susceptible to unauthorized use. The information stored at the information data stores may be stored locally at each of computing devices A–N and/or remotely (e.g., in a cloud computing environment provided by the external systems and which is accessible via the network of FIG. 1).

In operation, a user may request that their assistant schedule an event with one or more other users (e.g., user A may ask their assistant to “find and book a date that user B can join me for dinner.”) The requesting user’s assistant determines which user(s) are referenced by the request (e.g., user A’s assistant may determine that user B is referenced). In some examples, the assistant may determine which users are referenced by the request by comparing names included in the request with a contact list of the requesting user. In determining which users are referenced, the requesting user’s assistant may determine contact information for assistants associated with the referenced users. For instance, the contact list of the requesting user may
include a username of the referenced user along with universal resource locator (URL) or other address where the referenced user’s assistant can be contacted.

The requesting user’s assistant may communicate with the assistant(s) of the referenced user(s) to negotiate a time/date for the event (e.g., user A’s assistant may communicate with user B’s assistant to identify a time/date for user A to have dinner with user B). As negotiating the time/date requires that each assistant use their corresponding user’s data (i.e., calendar information), the assistants may perform the negotiation after receiving approval from their respective users. In some instances, the assistant of the requesting user may consider the receipt of the request to constitute approval.

The assistants may require approval for each new event negotiation. In some cases, a user may pre-authorize their assistant to negotiate event dates/times with one or more other users. As one example, an employee of a company can pre-authorize their assistant to negotiate meeting dates/times with other employees of the company. As another example, a user can pre-authorize their assistant to negotiate event dates/times with their spouse or significant other.

The assistants may negotiate the time/date using several methods. As one example, the requesting user’s assistant may provide the referenced users assistants with dates/times that the requesting user is available for the event (e.g., user A’s assistant may provide user B’s assistant one or more dates/time when user A is available to have dinner with user B) and the referenced users assistants may select one of the provided dates/times based on the referenced users’ availability (e.g., user B’s assistant may select one of the dates/times provided by user A’s assistant at-which user B is available to have dinner with user A). As another example, the referenced users assistants may provide the requesting user’s assistant with dates/times that the referenced user is available for the event (e.g., user B’s assistant may provide user A’s assistant one or more dates/time when user B is available to have dinner with user A) and the requesting
user’s assistant may select one of the provided dates/times based on the requesting user’s availability (e.g., user A’s assistant may select one of the dates/times provided by user B’s assistant at-which user A is available to have dinner with user B). As another example, all of the assistants may provide times/dates that their corresponding users are available for the event to a separate assistant or service which may process the availabilities and reply to the assistants with a date/time that all of the users (i.e., the requesting user and the referenced user) are available.

If none of the provided dates/times work (e.g., if user A’s availability does not overlap with user B’s availability), the assistant may inform the requesting user that the event could not be scheduled. In some examples, the assistants may propose moving previously scheduled events to accommodate the new event request (e.g., if user A has a conflict during every date/time offered by user B’s assistant, user A’s assistant may propose rescheduling one of the events scheduled during one of the dates/times offered by user B’s assistant).

With the date/time for the event agreed upon, the assistants may create entries in calendars of their corresponding users (e.g., user A’s assistant may create an entry in user A’s calendar for the dinner and user B’s assistant may create an entry in user B’s calendar for the dinner). The assistants of the referenced users may communicate with the assistant of the requesting user to obtain more detailed information about the event in order to create more complete calendar entries (e.g., user A’s assistant may output a request to user B’s assistant to book the agreed upon date/time, provide user B’s assistant with location and contact information for the dinner.

By automatically negotiating a time and date for an event with assistants of other users invited to the event, the assistant of the requesting user may reduce the scheduling burden of group events. The above examples are just some use cases for the assistant architecture shown in FIG. 1, the assistant architecture has many other applications and use cases.