July 2020

PROXIMITY BASED MEETING DIAL-IN

Konrad Reszka

Kevin Redmon

Bradford Ingersoll

PROXIMITY BASED MEETING DIAL-IN

AUTHORS:

Konrad Reszka
Kevin Redmon
Bradford Ingersoll

ABSTRACT

A solution is provided that leverages a digital proximity-based mechanism to determine that a meeting is taking place and then to automatically trigger a notification (with a close group of colleagues) so that others may also join the meeting remotely.

DETAILED DESCRIPTION

A common occurrence in any office is the impromptu meeting. Two or more people happen to get together to discuss a topic. What frequently occurs in larger offices, is that a third colleague notices this gathering and decides to join in as well. Quite often these meetings are the source of valuable training - discussing topics that the participants were not planning on asking.

Remote teams and users however often feel isolated and lack this ability to join impromptu meetings that frequent larger office sites. This causes them to not only feel disconnected, but to miss out on knowledge transfer, problem solving, and ideation.

Most collaboration applications allow a user to define a list of contacts. Not all of these contacts, however, are a part of that user’s social, local, or technical circle of collaborators. The smaller subset of contacts with which the user most frequently collaborates with (by working on the same project, being on the same team, or being physically located in the same area) can be marked as "favorite". It is these colleagues, these favorite contacts, which we pay attention to, notice when they gather nearby, and wish to join in discussions for both work and play. When we see three or more of these contacts gathered together across the physical office, they must be discussing something interesting. Remote users however do not have the luxury to see when these events occur, and thus miss out on knowledge transfer, ideation, and socialization. Furthermore, these unplanned, impromptu meetings, are not on calendars, they are not advertised, and since everyone is local - no video/voice sessions are started.

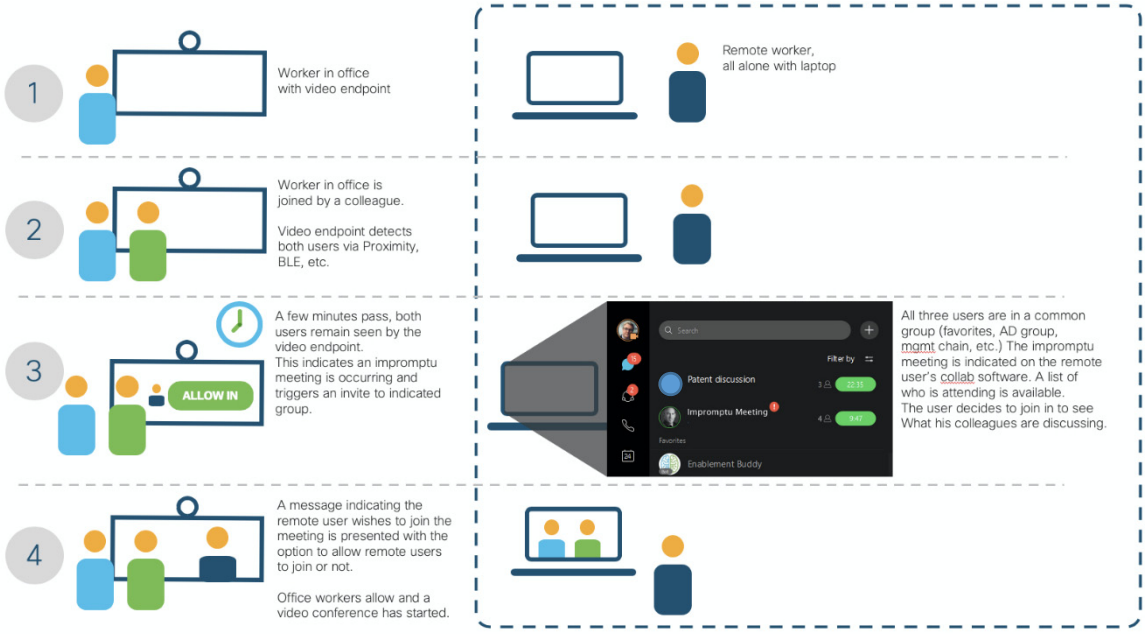

Figure 1

Collaboration endpoints are capable of identifying nearby users through the use of short distance discovery protocols (a proximity-based pairing technology, such as Cisco's Intelligent Proximity™ technology or Bluetooth®) communicating with corresponding applications installed on their personal phones, laptops, or tablets.

With reference to Figure 1 above, the following is an example process:

1. A collaboration endpoint continuously discovers nearby users.
2. If that endpoint is registered to a particular user, it has that users list of favorited contacts.
3. If more than two favorited contacts are discovered and present then:
    1. Start a video meeting session.
    2. Iterate the list of the user's favorited contacts.
        1. If the contact is one of the discovered users (physically present) then continue.
        2. If the contact has reciprocated the user as a favorite then notify that contact that a meeting has started and present the option for that

user to join/dial in to the meeting by voice or video (depending on the endpoint capabilities).

3. The endpoint would notify the user that there is an incoming call and a user wanting to join the meeting.
4. The user has the option to admit or decline the call request.

Starting voice/video sessions is never a consideration for impromptu physical meetings, nor is inviting other people who may or may not be interested. This approach is better because by combining proximity technologies with collaboration technologies, the entire process can be automated and ultimately be more inclusive for the whole team.

Further attributes may be used, such as timers, to fine tune what constitutes a meeting worth attending (for example to distinguish from brief comments when passing by) or to identify favorite contacts to ignore entirely in this system (such as contacts sitting within proximity range).

This solution can work on any device that has some endpoint discovery method. The location of the device is not relevant (i.e., public, personal office, conference room, etc.). The feature can be disabled on a per-endpoint basis ("Impromptu Collaboration" enabled/disabled or "Do Not Disturb" mode) and the creation of a meeting may be triggered at the source (do we want others to potentially join our meeting) as a "button" on the local screen (a per-meeting option). Acceptance of the collaboration meeting at the remote end may also be by way of a "button" on the screen to accept the invite. (i.e., walking away from your physical desk to join a collaboration meeting).

This may be based on a "Birds of a Feather" concept - if you see familiar faces in a meeting, you know what they are likely talking about and you want to join the discussion. The original device will trigger an invite to the mutual favorites. Each of the remote invitees will see those that have accepted (as established based on the logical AND of the original collaborators' mutual associates list as determined by their "Common Group", i.e. Favorites, Corporate Structure, etc.). As in a real-world experience, you may not always know what is being discussed across the room but you see familiar faces in the meeting and want to join in on the conversation. If the conversation is not relevant to you or you no longer have the time to collaborate, you can also exit and rejoin the conversation as you wish.

The Contact List component may include Contact List Options such as:

- Active Directory Group
- Organization Structure
- Local Contact/Favorite List
- Other

With the Contact List provided above, any number of "relationships" can be selected between the lists (such as a Logical AND of all attendees, close organizational relationships, AD group memberships, etc.). The goal of the contact list parsing for potential remote participants is to determine interest based solely on the people involved with the impromptu meeting.

A remote worker may learn know 'who' is in the impromptu meeting as follows. Within the Impromptu meeting pop-up, a user will be able to see the list of users that have already accepted the invite. Hovering over the user count of the Impromptu meeting, a user would see the other users that are present in the meeting, such as shown in Figure 2 below.

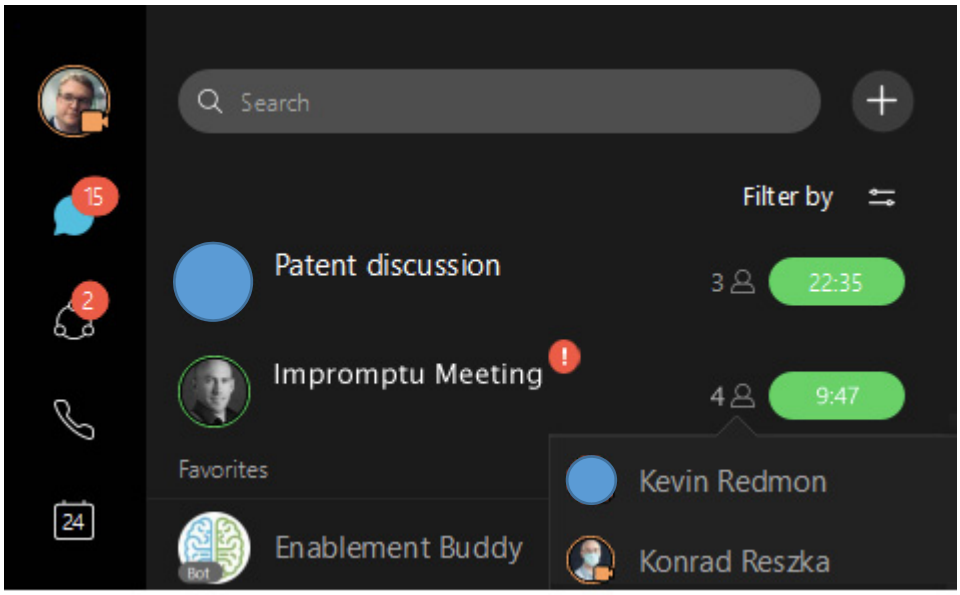

Figure 2

A remote user may not know - or care - about what the meeting is about. Historically, because of the limitations of the digital world, a remote user would not even know that an impromptu meeting is otherwise happening. Rather, remote users will be alerted of the meeting and can have the option to join or not join. The context of the meeting can be gleaned by knowing the attendee list. If the context is not relevant to them, the invitee can then exit and rejoin the meeting at will.

The foregoing solution does not rely on an auto-calling function. The feature will be triggered via a PUSH notification to the application. If the invitee wishes to NOT attend, they can simply ignore the push and/or put themselves on Do Not Disturb. As an added alert, any Impromptu meeting Invites would be shown in the GUI as long as they are ongoing so invitees can join and leave at will.

With regard to privacy/security issues, prior to any remote attendees being invited into the meeting, a prompt (via an on-screen button) can be offered on the local video conferencing unit. This is indicated above. If remote attendees are allowed by joining by default (in an "auto-accept" fashion), an audible prompt will be sounded (at minimum) with an affirmation of attendee joining. This would be analogous to someone walking into a real-world, impromptu conversation. For a private conference room or conversation, a "Do Not Disturb" button may be provided to disable this feature in an ad hoc manner. This Do Not Disturb feature would be akin to a physical door being closed on a meeting in a private conference room.

To summarize, this solution leverages a digital proximity-based mechanism (Bluetooth, Proximity, Wireless, etc.) to determine that a meeting is taking place and then to automatically trigger a notification (with a close group of colleagues) so that others may also join the meeting remotely. Using devices capable of short distance discovery, it is possible to detect and alert a user when 3 or more of that user’s contacts are discovered by a single video endpoint (a good sign that an impromptu meeting is occurring) to initiate a call to the local video endpoint to virtually join in the impromptu meeting.