

# Technical Disclosure Commons

---

Defensive Publications Series

---

December 08, 2017

## DYNAMIC NOTIFICATION SETTINGS FOR NEW OR UNUSED NOTIFICATION TYPES

Joseph Cohen

Justin Lewis

Follow this and additional works at: [http://www.tdcommons.org/dpubs\\_series](http://www.tdcommons.org/dpubs_series)

---

### Recommended Citation

Cohen, Joseph and Lewis, Justin, "DYNAMIC NOTIFICATION SETTINGS FOR NEW OR UNUSED NOTIFICATION TYPES", Technical Disclosure Commons, (December 08, 2017)  
[http://www.tdcommons.org/dpubs\\_series/917](http://www.tdcommons.org/dpubs_series/917)



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.

## **DYNAMIC NOTIFICATION SETTINGS FOR NEW OR UNUSED NOTIFICATION TYPES**

Websites and applications installed on computing devices (e.g., mobile devices, stationary devices, etc.) may use notifications to increase traffic to the websites and applications and/or to retain users. The websites and applications may include social network platforms, content sharing platforms (e.g., videos, audio, images, etc.), streaming services (e.g., video games, videos, audio), and the like. Subject matter categories associated with the websites and applications may include books, business, catalogs, education, entertainment, finance, food and drink, games, health and fitness, kids, adults, lifestyle, magazines and newspapers, medical, music, navigation, news, photos and video, productivity, reference, shopping, social networking, sports, travel, utilities, weather, and the like. A notification may refer to a message sent and/or displayed by the website or application that includes suggestions, recommendations, information, or the like. The notifications may include text, images, videos, audio, etc. A notification may be associated with notification settings that include a type and a state. If the state is set to on, then that type of notification may be sent and/or displayed. If the state is set to off, then that type of notification may not be sent or displayed.

To use notifications effectively, applications should balance consuming more of a user's attention with the risk that the user may ignore or disable the notifications if they are sent too frequently. As such, applications may provide users with a way to disable notifications that they are not interested in via a notification settings panel. Some notification settings panels display just the notification settings for the type of notification that the user has already received. Other notification settings panels may display every notification setting available for each website or application regardless of whether the type of notification is relevant to the individual user.

Just displaying notification settings for the types of notifications that the user has already

received may miss an opportunity to increase user interaction and satisfaction with notification types they have not yet received. For example, the user may be unable to consent to receiving a notification type that the user has not yet received, and the user may be unable to decline to receive a notification before they receive it. Statically displaying many or every notification setting for each website and/or application may inundate the user and increase the likelihood that the user misses notification types in which they may be interested. Accordingly, neither approach may increase the number of interesting notifications a user receives and decrease the number of disinteresting notifications the user receives.

We present a mechanism that solves these issues by dynamically populating a user settings menu for notifications (e.g., a notification settings panel, a notification settings toolbar, etc.) with new notification types and/or sources from which the user has not received a notification. The mechanism may display a finite number of notification settings to the user ordered partially by the likelihood that the user is to receive those types of notifications. The mechanism may display the set of notification settings for notifications the user has received in the past and also notification settings for notification types that the user is likely to receive in the future. As such, the user may enable notifications that interest them and that they may have been unaware of. Also, the user may disable the notification to prevent being spammed by unwanted notifications. Further, the mechanism may not display notification settings for notification types that the user is unlikely to receive or qualify for.

The notification settings panel may refer to a separate screen within a user interface of the website or application that includes the dynamically selected notification settings. The notification settings toolbar may refer to an embedded user settings menu where a portion of the notification settings are shown in an element (e.g., dialog window, toolbar) in a portion of the

user interface. The other portions of the user interface may display content of the website or application. In some instances, the notification settings toolbar may be dynamically inserted into the user interface of the website or application and may include a notification setting that is contextually relevant to a particular entity (e.g., content creator, channel, a contact associated with the user, etc.) engaged with by the user. The variables used to determine if a specific notification setting is visible in the notification settings panel may also be used to determine the visibility of the notification setting in the notification settings toolbar.

The mechanism may maintain and traverse a user event history for user events associated with a notification source and/or one or more notification types. When user events are identified, the mechanism may increase a user affinity score for the notification source associated with the identified user event. The notification source may include a content creator (e.g., artist, producer, etc.), a channel, an associated contact (e.g., “friend”), etc. that is associated with content provided via the website and/or application. The user events may include the user accessing, viewing, listening to, and/or editing content associated with the notification source for a threshold period of time, accessing, viewing, listening to, and/or editing a threshold amount of content associated with the notification source, selecting to subscribe to the notification source, subscribing to the notification source, consenting to receiving activity updates about the notification source (e.g., “following the notification source”), expressing approval (e.g., “liking”) of activity (e.g., comment, added content, shared content, etc.) by the notification source, requesting to create an association with the notification source (e.g., “friending”), and so forth.

There are a multitude of notification types for websites and/or applications, and it should be understood that any type of notification may be dynamically added or removed from a user settings menu using the disclosed mechanism. Some notification types may be per notification

source (e.g., content creator, channel, etc.) and may be associated with the user events described above. For example, the notification types may include notifications for the notification source uploading new content, the notification source broadcasting live content, the notification source livestreaming, the notification source making a social media post, the notification source commenting on a social media post, the notification source liking a social media post, the notification source uploading new content, and so forth. For example, a user event that includes watching a video on a notification source (e.g., channel X) may be associated with one or more notification types, such as receiving notifications about any videos uploaded by the notification source (e.g., channel X), receiving notifications when the notification source (e.g., channel X) livestreams, and so forth.

The mechanism may determine when the user affinity score for the notification source exceeds a threshold and may also check the state of the one or more notification types associated with the user event. The notification state may include three settings: on, off, and “not set by the user”. The notification state of “not set by the user” indicates that the user has not explicitly set this notification setting and has not previously interacted with this notification setting before. If the notification state is “not set by the user” for the one or more notification types, and the user affinity score exceeds the threshold, then the mechanism may preemptively add the notification settings having the one or more notification types for the notification source to the user settings menu in advance of the user actually receiving the notification. The mechanism may automatically set the one or more notification states to on (e.g., enabled) or may automatically set the one or more notification states to off (e.g., disabled) and may recommend that the user enables the one or more notification settings.

In one example, a social network platform may add a notification setting to allow the user

to consent to receiving every livestream when they become available if the mechanism determines that the user watches a lot of livestreams on the social network platform. In another example, the social network platform may use the mechanism to offer more event reminder notifications when a user misses an event (e.g., as indicated by his location or a last minute change in RSVP).

In an example, if a user expresses approval of a certain number of images associated with beaches on a content sharing platform, the mechanism may be used to add notification settings that allow the user to receive popular beach images. Further, if a user often checks their notifications for messages from a specific user on the content sharing platform, the mechanism may add a notification setting to allow the user to receive text or emails for messages from that user. Further, the content sharing platform may use the mechanism to suggest notification settings for new releases of content in various playlists and genres of music. In an example, if a user listens to a certain genre of music often, the content sharing platform may add a notification setting that allows the user to consent to being notified of new releases in that genre or newly popular songs in that genre. The mechanism may also add a notification setting that allows the user to consent to receive notifications related to local concerts and events for that genre.

In another example, streaming services can use the mechanism to add a notification setting that allows the user to consent to receiving notifications about new releases in genres they have watched or added. If a video in a user's watchlist is removed from the streaming service before the user has a chance to watch the video, the mechanism may be used to add a notification setting that allows a user to consent to receiving a notification for when content in the user's watchlist is about to be removed from the streaming service. Another example may include the mechanism adding a notification setting that allows users to consent to receiving notifications

about gamers that are rising in popularity on a streaming service, a new game that the user may be interested in watching, and the like.

The mechanism may be implemented as a service hosted by a server. The service may determine and maintain the notification settings that are to appear in the user settings menu of a user interface on a client device (e.g., mobile device, stationary device). The mechanism may determine that one or more notifications are candidates for a user based on relevancy (e.g., user affinity score for the notification source is above a threshold) and/or whether the user has seen the one or more notifications before (e.g., the notification state is set to “not set by the user”). When the mechanism determines the one or more notifications are candidates, the mechanism may add the candidates’ notification settings to a data store that is accessible by the client device. The dynamically generated set of notification settings that are tailored for each user may be retrieved by the client device and displayed in a user interface of the website and/or application on the client device.

The mechanism may also regularly update the set of notification settings for the various users. In one instance, the mechanism may remove notification settings that were preemptively added but are no longer relevant. For example, the user affinity score may be time dependent and when the user stops engaging a notification source (e.g., stops watching videos on a channel), the notification may become no longer relevant. In another instance, the mechanism may automatically disable notifications that the user has not interacted with. For example, if a notification is displayed to a user and the user consistently dismisses the notification, then the mechanism may disable the notification by setting its state to off but may still retain the notification setting in the user setting menu.

In another example, the mechanism can also be used to update the notification settings by

adding new notification settings based on the user's watch or engagement history with content. For example, if a user is watching content provided by a channel and has consented to receiving some notifications (e.g., newly uploaded videos for the channel) but not every notification for that channel, the mechanism may dynamically insert a notification setting in the user setting menu to allow the user to consent to enabling notifications that have not yet been consented to when events occur related to those notifications. To illustrate, the user may not have consented to receiving notifications for every livestream from the channel, and the channel may begin livestreaming, which may cause the mechanism to add a notification setting to the user setting menu to allow the user to consent to receiving notifications for every livestream from the channel.

Figure 1 depicts a flow diagram of a method for a client device to display a user settings menu in a user interface of a website and/or an application that includes dynamically generated notification settings provided by the mechanism, in accordance with some implementations. First, at step 102, a selection is received via a user interface of a website or an application displayed on the client device to perform a user event related to a notification source. As described above, the user event may include the user accessing, viewing, listening to, and/or editing content associated with the notification source for a threshold period of time, accessing, viewing, listening to, and/or editing a threshold amount of content associated with the notification source, subscribing to the notification source, consenting to receiving activity updates about the notification source (e.g., "following the notification source"), expressing approval (e.g., "liking") of activity (e.g., comment, added content, shared content, etc.) by the notification source, requesting to create an association with the notification source (e.g., "friending"), and so forth. The notification source may include a content creator, a channel, a

contact associated with the user, and so forth.

At step 104, the user event may be sent to the mechanism (e.g., service). The mechanism may maintain a user event history for each user that uses the website and/or the application. The mechanism may traverse the user event history to identify user events that may be associated with the notification source and/or with one or more notification types. When the user events are identified, the mechanism may increase the user affinity score for the notification source associated with the identified user event. The mechanism may determine whether the affinity score is above one or more thresholds and check the state of the one or more notification types associated with the user event. When the affinity score is above a first threshold for including the notification setting having the one or more notification types but below a second threshold for actually sending the notifications, and the notification states are set to “not set by the user,” the mechanism may add the one or more notification settings to the data store. In such a case, the state of the notification settings may be automatically set to off so the user can see the notification settings but does not actually receive the notifications. In another example, when the user affinity score exceeds both the first and the second threshold, and the states are set to “not set by the user,” the mechanism may add the notification settings to the data store. In such a case, the state of the notification settings may be automatically set to on so the user can see the notification settings and actually receive the notifications.

At step 106, a request is received to access a user settings menu (e.g., notification settings panel, notification settings toolbar). In some instances, the request may be received by the user clicking a settings button or link in the user interface, which may display the notification settings panel. In another instance, the request may be received when the user is viewing a user interface for an entity presenting content and the notification settings toolbar is dynamically inserted with

contextually relevant notification settings for the notification source and/or content.

At step 108, notification settings may be received from the mechanism. In one implementation, a data store may be accessed that includes notification settings provided by the mechanism. The notification settings may include at least one notification type that has not been received or at least one notification setting associated with a notification source for which a notification has not been received. At step 110, the user settings menu including the notification settings may be displayed. The notification settings may be relevant to the user (e.g., based on the user affinity score) and may be selected in advance of the user even seeing a notification associated with the notification type (e.g., based on the state of “not set by the user”). As such, the mechanism enables presenting a finite number of notification settings to the user that the user is likely to receive in the future. Further, since a threshold is used when evaluating the user affinity score for the notification source, certain notification types with user affinity scores below the one or more thresholds may be excluded from the user settings menu.

At step 112, user interaction with the notifications and/or the notification settings in the user settings menu may be received. For example, the user may click-through on the notification or may dismiss the notification. Also, the user may disable the notification in the user settings menu by setting its state to off or may enable the notification in the user settings by setting its state to on.

At step 114, the user interaction may be sent to the mechanism. If the user interaction indicates that the user does not interact with the notification because of dismiss rate, the mechanism may disable the notification setting by setting its state to off in the data store. In some implementations, the notification setting may be removed from the data store such that it is no longer displayed in the user settings menu. If the user interaction indicates that the user

interacts with the notification, then the mechanism may maintain the on state of the notification setting for the notification.

At step 116, the user settings menu may be updated with changes to the notification settings received from the mechanism based on the user interaction. In some implementations, the data store may be accessed to obtain changed notification settings based on the user interactions.

The mechanism described herein may enable retaining users of websites and/or applications and/or increasing traffic to the websites and/or applications. For example, the mechanism may dynamically populate a user settings menu with notification settings that are determine to be relevant to the user based on a user affinity score and/or based on whether the user has seen received the notification already or not. For example, the mechanism may include a notification setting preemptively in the user setting for notifications that the user is likely to receive in the future. This may allow the user to disable the notification prior to being spamming by the notification or enable a notification that the user has not seen yet but that interests the user.

## ABSTRACT

A mechanism for dynamically populating a user setting menu for notification settings is presented herein. The mechanism may present notification settings for new notification types and/or notification sources from which a user has not received a notification. The notification settings may be dynamically added to a user settings menu based on a user affinity score to a notification source and/or whether the notification type has been received by the user.

Accordingly, the mechanism enables displaying notification settings to the user that the user is likely to receive in the future prior to the user actually receiving the notification. This may enable the user to preemptively enable the setting or disable the setting to prevent spam.

**Keywords:** notifications, affinity score, notification settings, content, user settings menu, dynamic populating

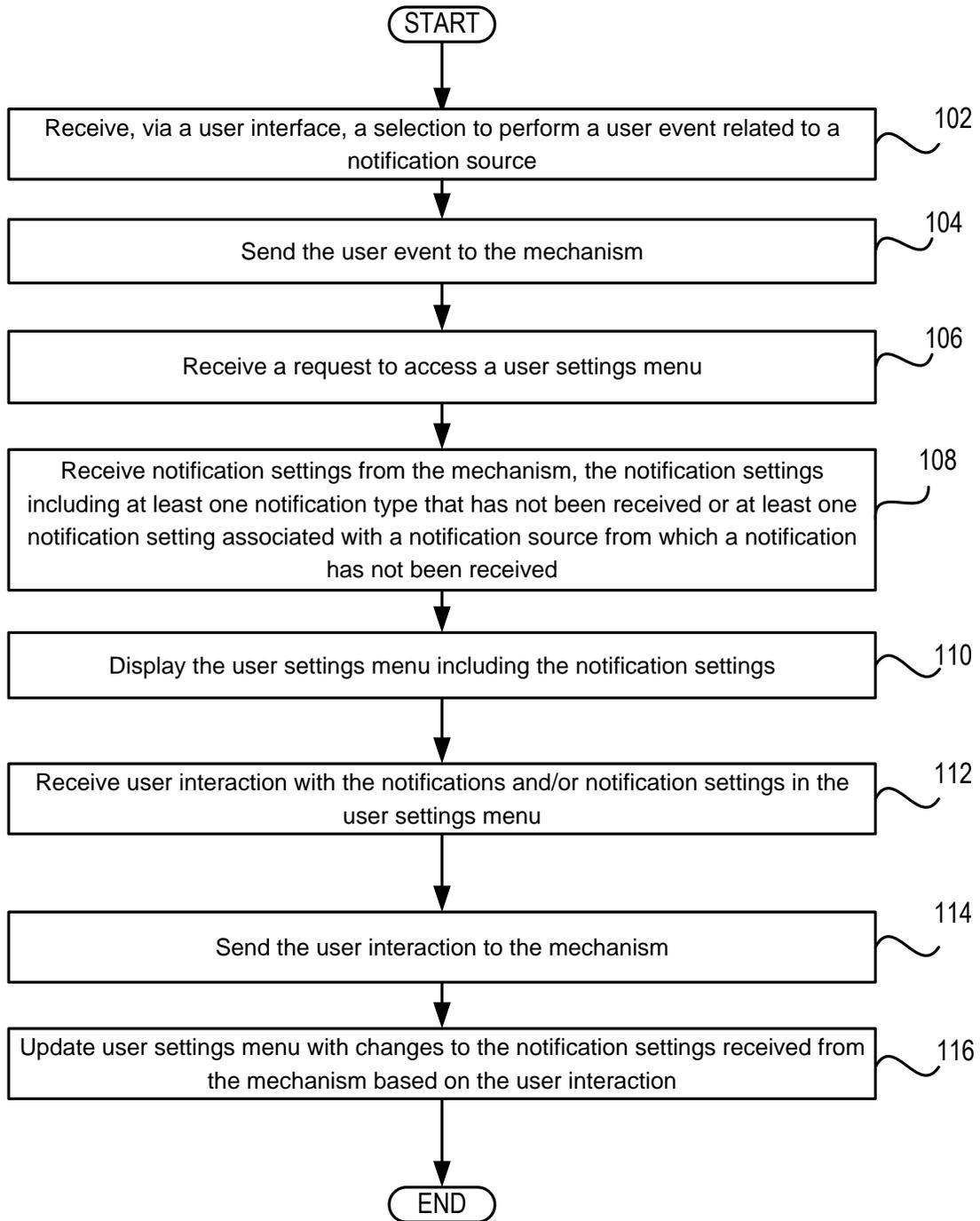


FIG. 1