

Developing for Android Wear - Part 1 (Bridged Notifications)

Justin Munger
GDG Chicago
July 8, 2014

Whoa? What do you mean Part 1?

- Android Wear SDK officially announced and launched at Google I/O two weeks ago
- Devices (for non-attendees) became available yesterday
- There are a lot of new concepts/information to digest and experience to be had
- Saving contextual notifications, apps and syncing for Part 2 in August

What is Android Wear?

- Wearable computer in a watch form factor
- Runs Android OS 4.4W (API Level 20)
- Pairs to phone/tablet through Bluetooth
- Uses Android Wear app and Google Play Services to sync between phone/tablet and Wear device
- Current devices available (as of July 2014) are LG G Watch and Samsung Gear Live

Why Android Wear?

- Removes the distractions caused by the immersive nature of mobile devices
- Delivers only the essential information needed in a timely manner
- Allows partial interaction with phones/tablets to perform certain necessary actions

Creative Vision for Android Wear

- Launched automatically
- Glanceable
- Suggest and Demand
- Zero or Low Interaction

Android Wear Design

Suggest - The Context Stream

- Vertical list of cards
- Cards display contextually-relevant (time- and/or location-based) information
- One card displayed at a time
- Optional background photos give additional context
- Cards pushed to user by Android Wear device and/or paired phone/tablet

Demand - The Cue Card

- Allows user to interact with Wear device through voice or tapping suggested options
- Contains pre-defined voice commands that applications can hook into
- Also allows for custom voice commands to start functionality
- Similar to how a person interacts with Google Glass and even phones/tablets

Notifications

- Part of the Suggest function of Android Wear
- Lives in the context stream
- Two types of notifications
 - Contextual notifications
 - Bridged notifications

Contextual Notifications

- Generated on the Android Wear device
- Uses current user context (time/location/sensor input) to display notifications to user
- Allows for more complex notification interaction (such as taking action directly on a card)
- More about this coming in Part 2!

Bridged Notifications

- Display of existing phone and tablet notifications on Android Wear
- Notifications are formatted differently than phone tablet
- Additional extensions to notifications allow wearable-specific features to be added
- This will be the subject of this presentation

How to Get Notifications on Wear

- Create notifications the same way that you would for phones and tablets
- Use NotificationCompat and NotificationCompatManager to create and post notifications
- Contain special add-ons for Wear
- Part of the Android Support Library (version 20.0.0 and above)

Code for Presentation

- Remainder of presentation demonstrates notifications with code examples from notification demo app
- Starts with basic notifications and builds upon them
- Code is available on GitHub (link at the end of presentation)

Bridged Notifications

How to Create a Notification

- Get an instance of `NotificationCompatManager`
- Create a Notification using `NotificationCompat.Builder()`
- Call `notify()` on `NotificationCompatManager`, passing in
 - Unique notification ID
 - Notification object

Bare-Bones Notification

- The minimum content required for a notification is a small icon (24x24dp)
- Takes a drawable resource ID
- Centered on upper-right side of card
- Code:

```
Notification notification =  
    new NotificationCompat.Builder(getActivity())  
        .setSmallIcon(R.drawable.notification_icon)  
        .build();
```


Basic Notification

- Contains the basic elements that typical notifications have:
 - Small icon
 - Content title
 - Context text
- Title and text are left aligned on the card

Notification with Large Icon

- Provides background image for notification
- Adds additional context to notification
- Takes a Bitmap image (use BitmapFactory.decodeResource())
- Shown once card is swiped upwards from peek state

Notification with BigTextStyle

- Allows display of long text (email messages) in a notification
- Displays enough content to fill screen
- Tapping on card expands entire text content
- Create NotificationCompat.BigTextStyle and use bigText() method to add text
- Call setStyle() on Notification to set text to notification

Notification with Content Intent

- Allows user to start an intent on the phone/tablet
- Same effect as user tapping notification on phone/tablet
- Creating the Intent and wrap it in a PendingIntent
- Adds page with “Open on phone” action button to notification

Notification with Action Button

- Adds page to notification with button/text
- Icon image is placed inside of a blue circle
- Text is displayed under the button
- Intent (wrapped in PendingIntent) sets action for the button to perform on phone/tablet
- Uses NotificationCompat.Action to define action
- Added with addAction() to the Notification

Wear-Only Action Button

- Same as notification with action button but action button only displays on Wear device
- Uses `NotificationCompat.WearableExtender` to set Wear-specific features for notification
- `WearableExtender` is added to `Notification` using `extend()`

Notification with Voice Reply

- Uses RemoteInput.Builder to construct a voice reply input screen
- NotificationCompat.Action.Builder creates the action button that starts the voice input
- Action is added to WearableExtender
- WearableExtender is added to Notification

Notification with Multiple Pages

- First notification is prepared using `NotificationCompat.Builder`, but `build()` is not called
- Subsequent notifications are created as other previous notifications have been
- Final notification adds subsequent notifications using `addPage()`
- First notification added to final using `extend()`

Stacked Notifications

- Create notifications as the previous notifications have been
- Assign each notification to a group by calling `setGroup(GROUP_ID)` where `GROUP_ID` is a string identifier
- Display each notification individually and they will be stacked together automatically
- Does not display on phone/tablet

Stacked Notifications (Inbox-style)

- Allows stacked notifications to appear on phone/tablet
- Create stacked notifications as before
- Create `NotificationCompat.InboxStyle` and populate with title, summary text and content
- Create new `Notification` and `setStyle()` with `InboxStyle`, associate with group and indicate notification is part of a group

Questions?

Additional Resources

- Android Wear Training: <http://developer.android.com/training/building-wearables.html>
- Google I/O Dev Bytes Wearables Videos: <http://goo.gl/yqWsq4>
- Demo code: <http://github.com/justinkmunger/DevelopingForAndroidWear>

Thank You!