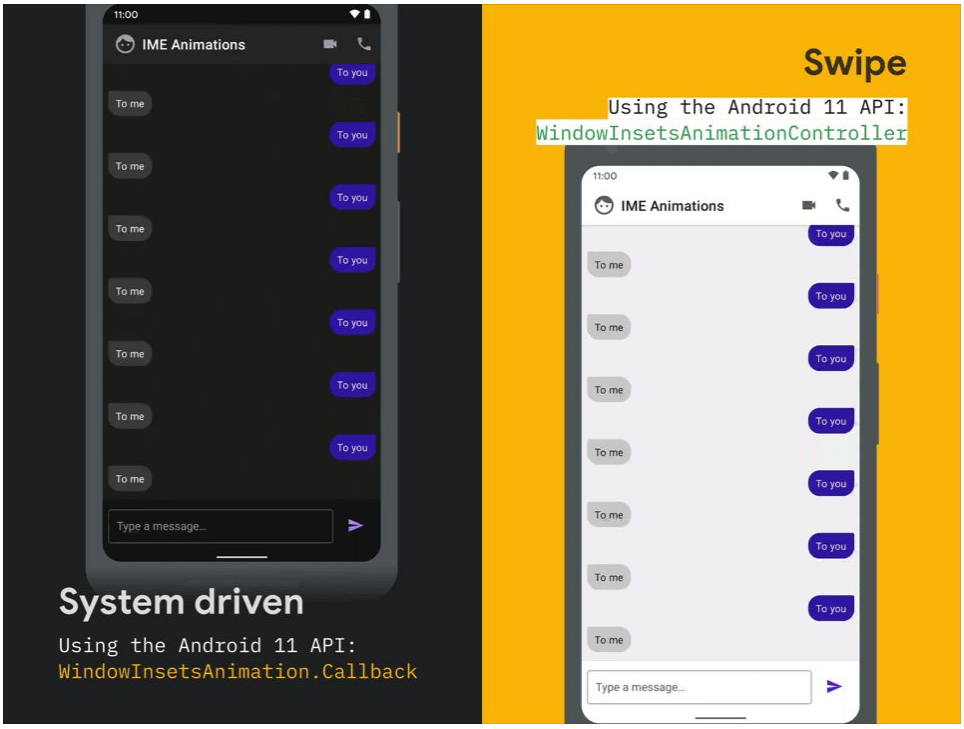
What's new in Android 11

1. UI
    a. Window Insets - More information about the multiple types of content being displayed
        i. Such as Informations about Status, Navigation, IME
        ii. Many methods in the WindowInsets deprecated, Window-type specific things are added.
        iii. Via listener, we can get the changes of status on insets.
    b. IME Animations
        i. Synchronize keyboard animations with app content changes
        ii. Listener for changes
        iii. Drive keyboard animation directly / or just leave it as system-driven
        iv. Due to the parameter we put into the windowInsetAnimation, we can see it works differently
            1. Setting duration
            2. LinearMotion (interpolator)
            3. Cancellation allowed
            4. Animation control listener
            5. editText.setWindowInsetsAnimationCallback(WindowInsetsAnimation.Callback)
    c. 

    d. Conversations
        i. One of system UI area / one of way to interact with users
        ii. Via existing notification mechanism
        iii. Bubble upcoming next is a way to converse.

        iv. We can change the priority of the conversations and choose how the system should notify us.
        v. Person, ShortcutInfo and Notification
    e. Bubbles
        i. Like a facebook chat icon on the home screen
        ii. If an user taps that bubble then it will show a mini activity
        iii. Notifications that can also show as bubbles
        iv. In Android 10: Developer option
        v. Better than the System Alert window.
        vi. Created with Notification API with more metadata and dedicated activity
        vii. Activity corresponding to this bubble should be declared in Androidmanifest.xml
        viii. Notification.BubbleMetaData, shortcut Id
2. Privacy
    a. Data Access Auditing
        i. When app code accesses private data
        ii. Code in a third party or external library accesses private data
        iii. Registering callback, and this callback is triggered when it occurs
        iv. AppOpsManager, AppOpsManager.OnOpNotedCallback
    b. One-Time Permissions
        i. The permission that the user only can be granted at that time.
    c. Background Location
        i. More restrictive in Android 11
        ii. At first request foreground permission
        iii. Then request background permission -> Take user to Settings and let him/her trigger the permission manually
    d. Foreground Services
        i. Location in Android 10
        ii. Camera, Microphone in Android 11
        iii. Declaring attributes in AndroidManifest.xml file
    e. Package visibility restrictions
        i. It’s not possible to query all of the packages installed on a device
        ii. Instead, we need to declare in the manifest what we want to access.
    f. Scoped storage
    g. Auto-reset permissions
        i. Dealing with the particular case that an user hasn’t used an app for a couple of years since he/she installed and ran the app, granted permission.
        ii. Then the system automatically reset the permission granted
3. Developer Goodies
    a. Wi-Fi Debugging - Android Studio 4.2 Canary
        i. Manually enable that option, pair and connect with the device

b. Nullability Annotations - it has been added to platform APIs over the last couple of years
    i. @RecentlyNullable, @RecentlyNonNull
        1. It indicates that it has just added them recently, and when built it shows warning.
    ii. @Nullable, @NonNull
        1. Errors - when passing null to nonnull annotated parameter
c. Crash Reasons Reporting
    i. API to query why your app crashed.
    ii. Crash Reasons Querying - with getHistoricalProcessExitReasons() in activityManager which returns List of ApplicationExitInfo
d. GWP-ASan
    i. For native code/libraries
    ii. Android 10: HWASan
    iii. Catches memory issues
    iv. On user devices in the field
    v. Low overhead (runtime and memory)
    vi. Reports uploaded to Play dashboard
    vii. Declaring in AndroidManifest.xml
e. ADB Incremental
    i. For huge APKs (like Game)
    ii. Faster installs via command-line (up to 10x faster)
    iii. First: sign APK, create APK signature scheme v4 file
    iv. Adb install --incremental
f. Behavior Changes
    i. Most changes only take effect on targetSDK R
    ii. We can test changes with behaviour toggles when we want to target, if we want to do.
    iii. Command Line vs UI which is newly exposed

4. Graphics
    a. NDK Image Decoders
        i. All decoders are directly accessible to native code (JPEG, GIF, PNG, WEBP...)
        ii. No more JNI up-calling and down-calling to Android SDK
        iii. No more Bundling decoder libraries
        iv. No more bulky apk by the bundles
    b. Animated HEIF
        i. We can load animated images from HEIF files as AnimatedImageDrawables, using ImageDecoder
        ii. Smaller than GIF
        iii. Should be performed off the main thread when decoding.
    c. NDK: OpenSL ES (Deprecated)
        i. Oboe for the win! (Open source)

        ii. CPP library providing high performance audio features
    d. Variable Refresh Rate
        i. With the rise of high refresh rate, Surface.setFrameRate() can enable us to set the frame rate, especially if we have our own rendering loop in the app.
5. Furthermore
    a. Neural Networks API
        i. For ML stuff
        ii. C API for on-device ML with version 1.3 along with Android 11
    b. 5G API
        i. APIs to optimise 5G experiences
            1. Metered network state
            2. Bandwidth estimate
            3. A callback is in ConnectivityManager, which returns the state of the network and lets the user cope with the changes.
    c. Autofill / Keyboard Integration
        i. Autofill content in keyboard, not drop-down
        ii. InputMethodService
        iii. Keyboards and Password apps
6. Non-Platform
    a. Jetpack
        i. 70+ libraries, releases every two weeks
        ii. Hilt (DI built on the top of Dagger)
        iii. Paging 3.0 - All Kotlin with Coroutines
        iv. CameraX beta
        v. Further discussion in the other session about Jetpack
    b. Jetpack Compose
        i. new UI toolkit for Android
        ii. More on the other session about Jetpack Compose
    c. Android Studio
        i. 4.0 stable
            1. Motion editor (for MotionLayout)
            2. LayoutInspector
        ii. 4.1 beta
            1. Database Inspector (Room. SQLite)
        iii. 4.2 Canary
            1. Wireless debugging with Android 11
            2. Jetpack Compose development
        iv. More on the other session about Dev Tool and Design Tool
7. Google Play
    a. New Play Console in beta now
        i. Complete redesign
        ii. Clearer and easier to use

b. What’s new in Google Play