





# Monetizing Android Apps

Chrix Finne  
Kenneth Lui



## 3 things to think about



User engagement



Distribution



Monetization



# 3 sessions + 1 code lab on building an app business

## Monetize

Monetizing Android Apps

Day 1 @ 14:45 - 15:45

## Promote

Google Play:  
Marketing 101 for  
Developers

Day 1 @ 16:00 - 17:00

## Measure

Measuring the  
End-to-End Value  
of Your App

Day 3 @ 11:30 - 12:30



# 3 sessions + 1 code lab on building an app business

## Monetize

Monetizing Android  
Apps

Day 1 @ 14:45 - 15:45

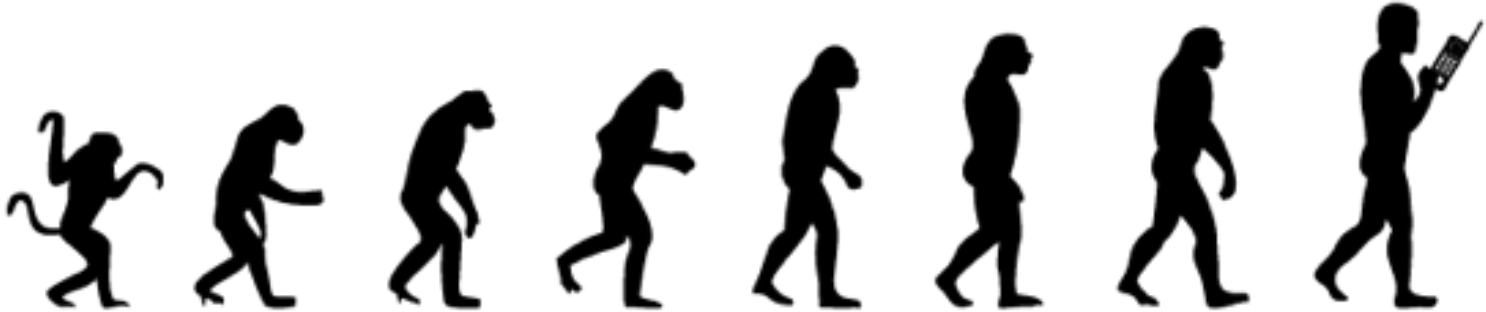
{Code lab}

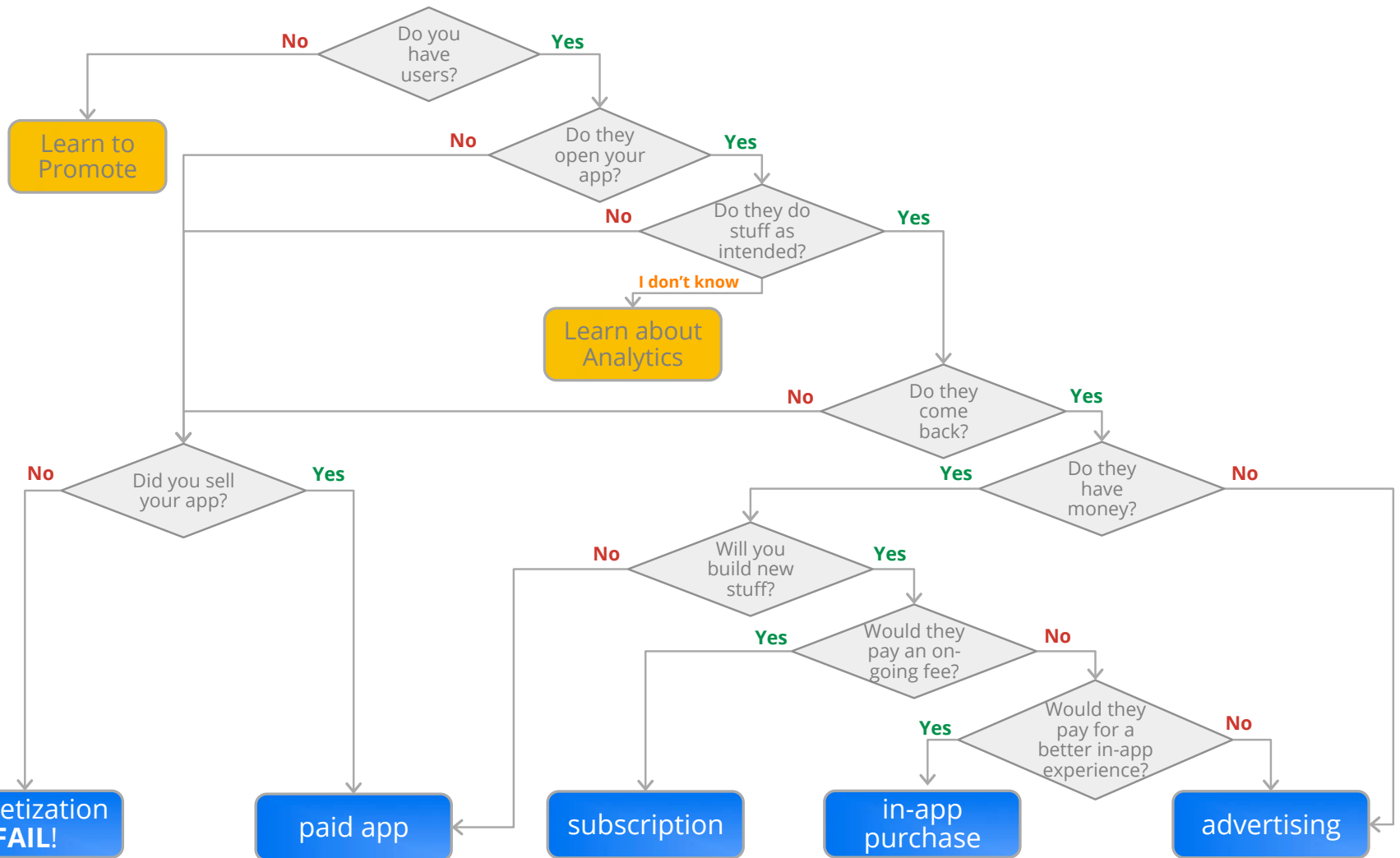
**Android App Monetization**

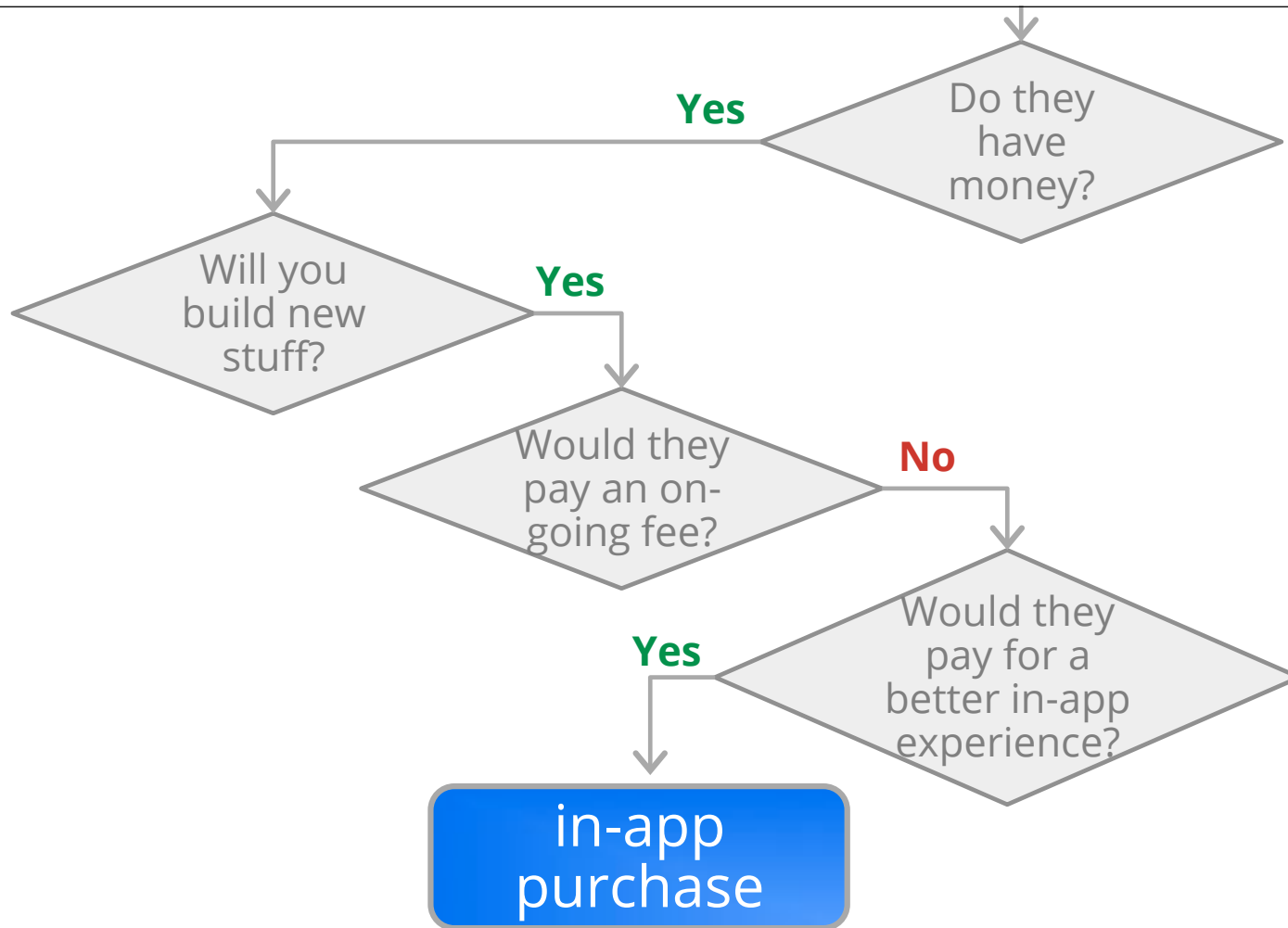
Day 2 @ 11:45 - 13:30



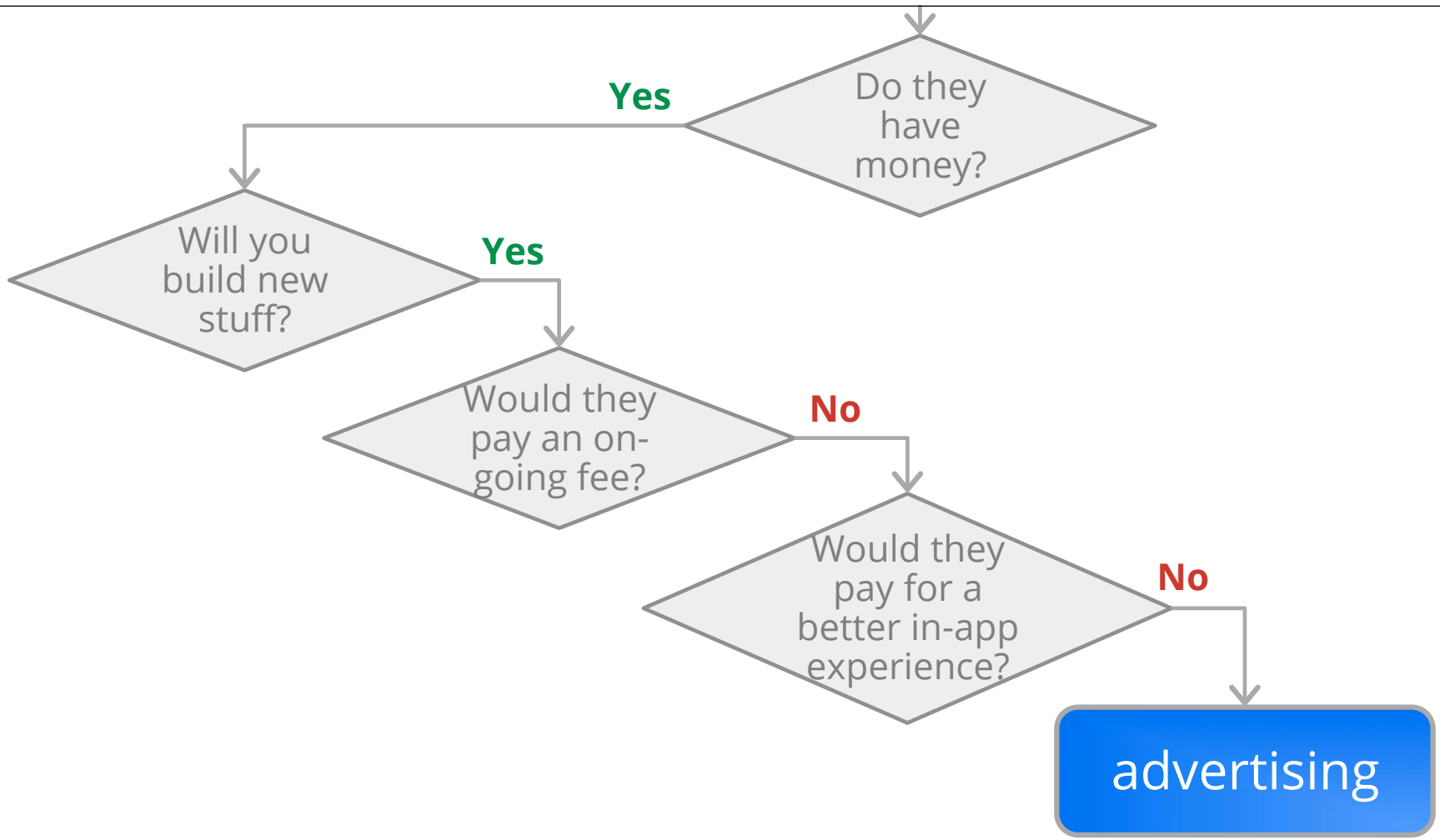
# Evolution of app monetization











# Make money with Google



**Inside** Google Play store



**Outside** advertising



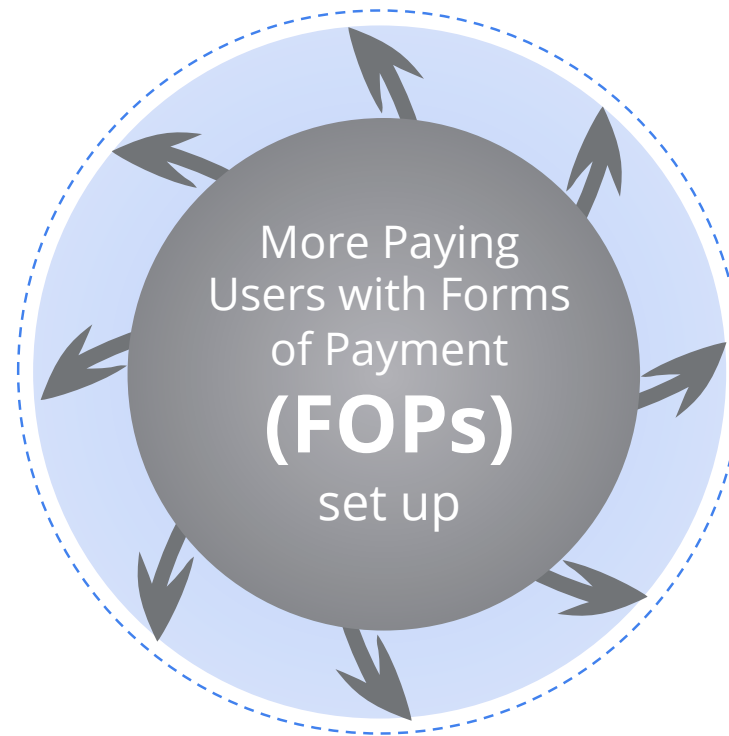


# Making money with Google Play

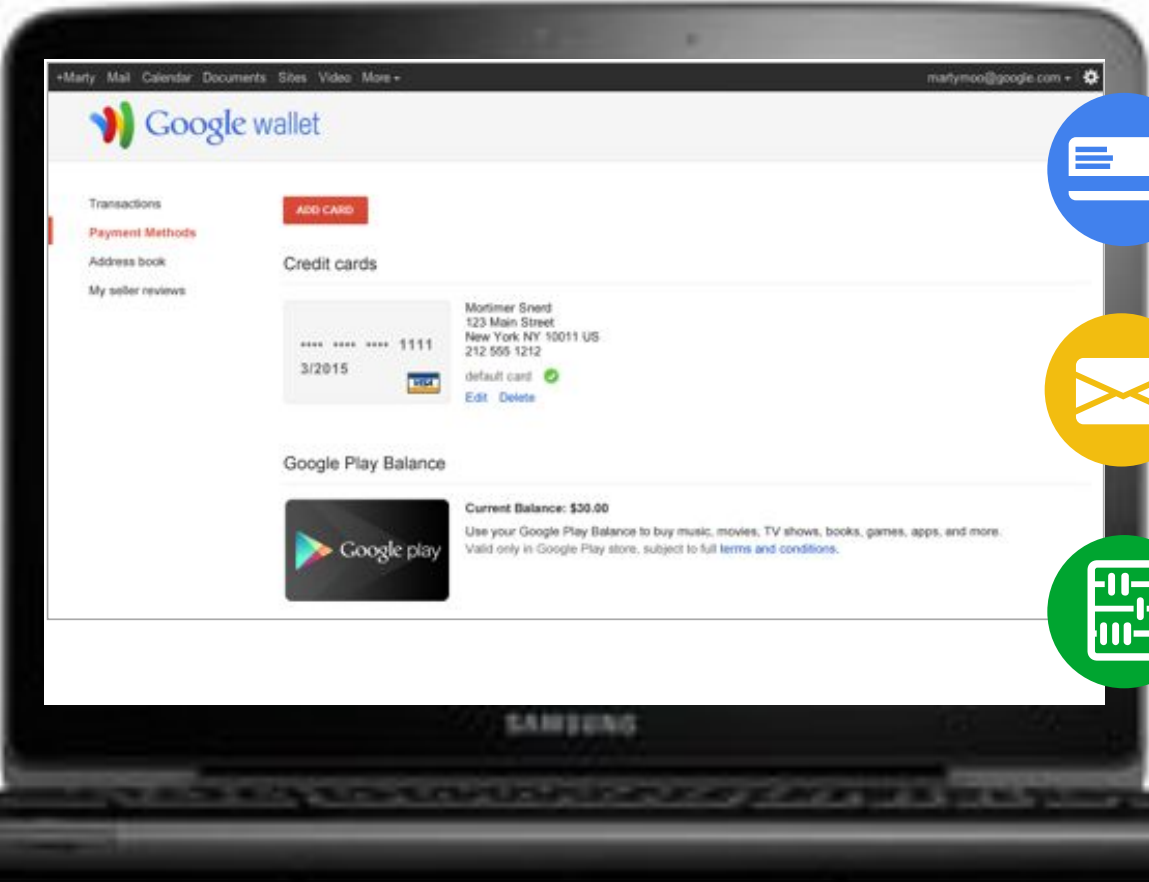
# Expanding Monetization Opportunities



# Expanding Monetization Opportunities



# Forms Of Payments (FOPs)



Credit / Debit Cards



Direct Carrier Billing

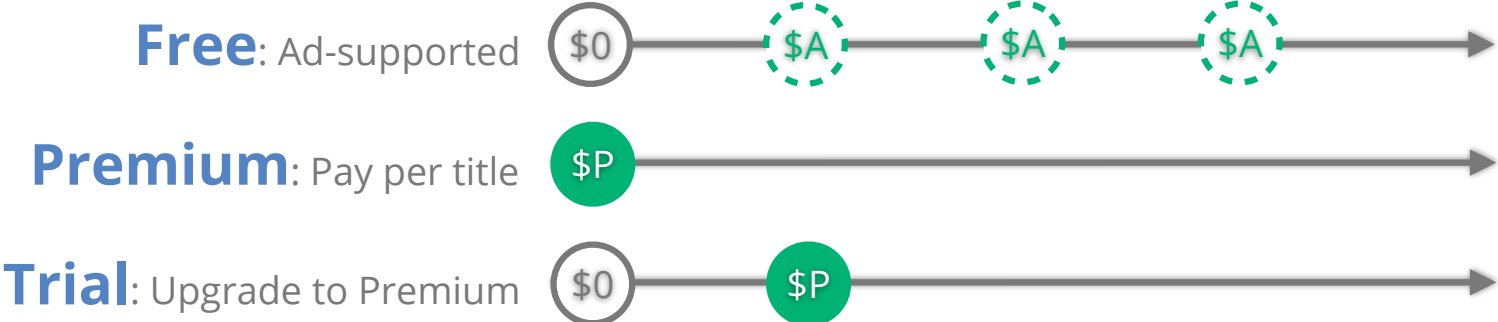


Google Play Balance

# Expanding Monetization Opportunities

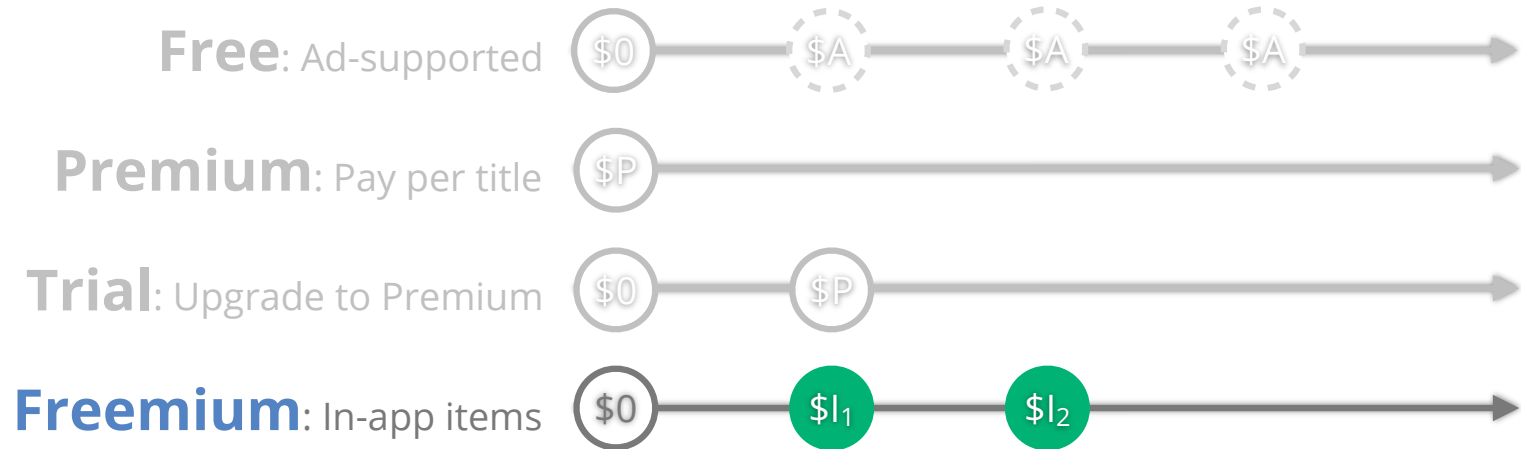


# Monetization Options

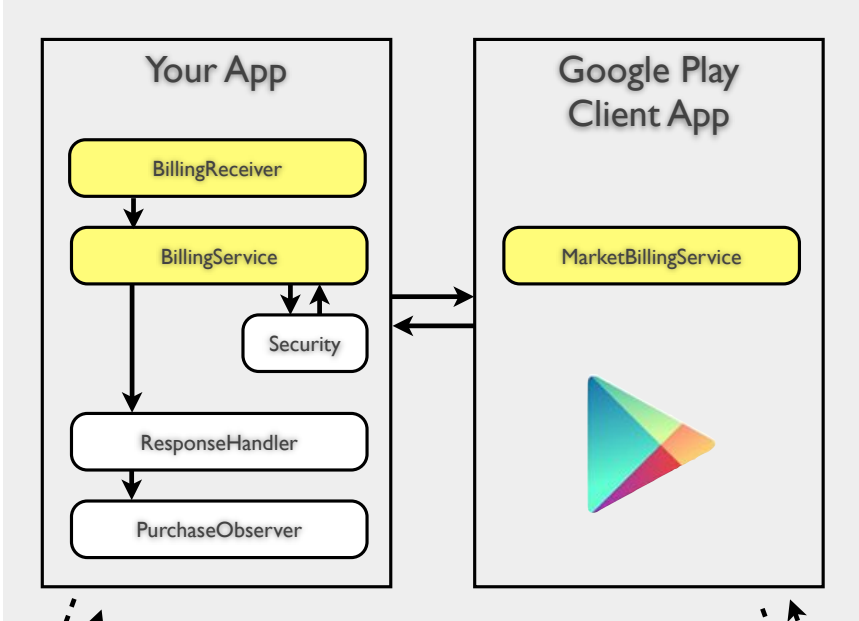




# Monetization Options



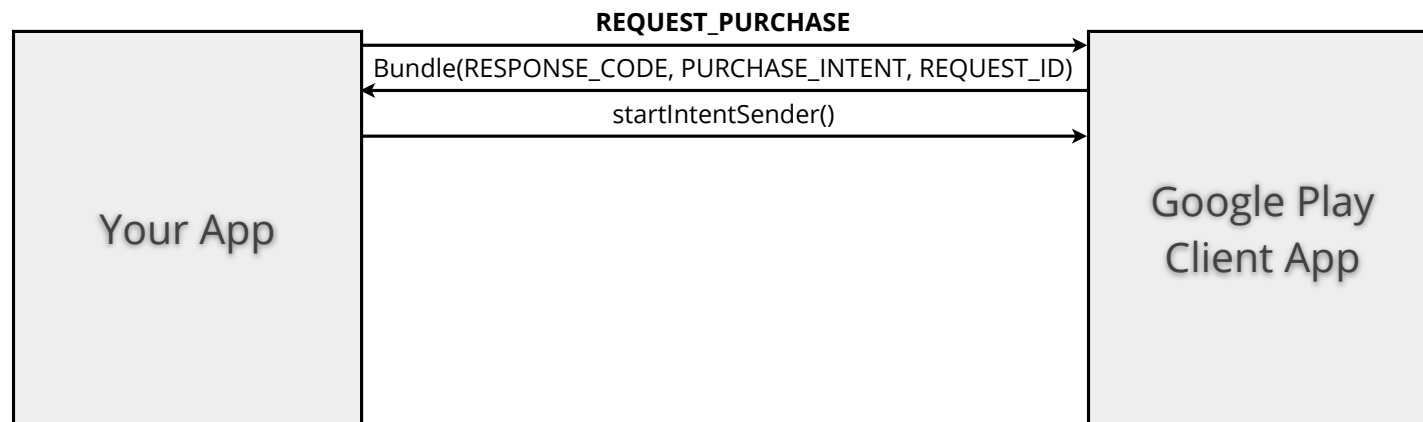
# In-app Billing Overview



Two types of IAP  
1. Managed  
2. Unmanaged



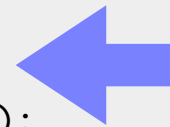
# In-app Billing Overview - Requesting a Purchase



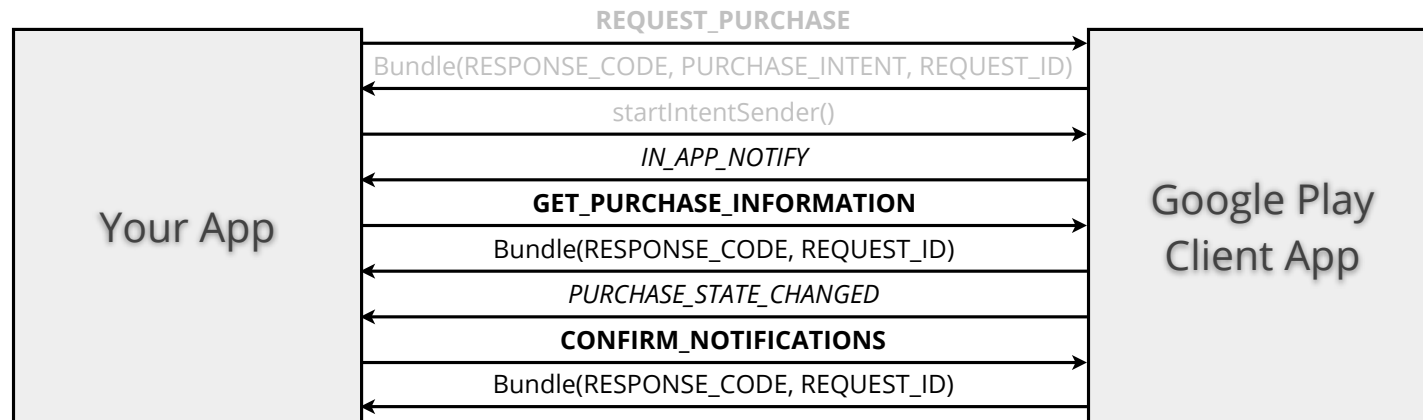
# In-app Billing Overview - Requesting a Purchase

```
protected long run() throws RemoteException {  
  
    Bundle request = makeRequestBundle("REQUEST_PURCHASE");  
    request.putString(Consts.BILLING_REQUEST_ITEM_ID, mProductId);  
    request.putString(Consts.BILLING_REQUEST_ITEM_TYPE, mProductType);  
  
    Bundle response = mService.sendBillingRequest(request);  
    PendingIntent pendingIntent =  
        response.getParcelable(Consts.BILLING_RESPONSE_PURCHASE_INTENT);  
  
    Intent intent = new Intent();  
    ResponseHandler.buyPageIntentResponse(pendingIntent, intent);  
  
    return response.getLong(Consts.BILLING_RESPONSE_REQUEST_ID,  
        Consts.BILLING_RESPONSE_INVALID_REQUEST_ID);  
}
```

Java



# In-app Billing Overview



```
{ "nonce" : 1836535032137741465,
  "orders" :
  [{ "notificationId" : "android.test.purchased",
    "orderId" : "transactionId.android.test.purchased",
    "packageName" : "com.example.dungeons",
    "productId" : "android.test.purchased",
    "purchaseTime" : 1290114783411,
    "purchaseState" : 0,
    "purchaseToken" : "rojeslcdyyiapnqcyнкjyyjh" }]
}
```



# Freemium Model - Case Study

## Tiny Village by TinyCo

Launched in December 2011

Free-to-Play, with in-app purchases

4.5 star rating; >3M downloads

## Strategy

Engage users

Convert to purchasing players

Monetize paying users

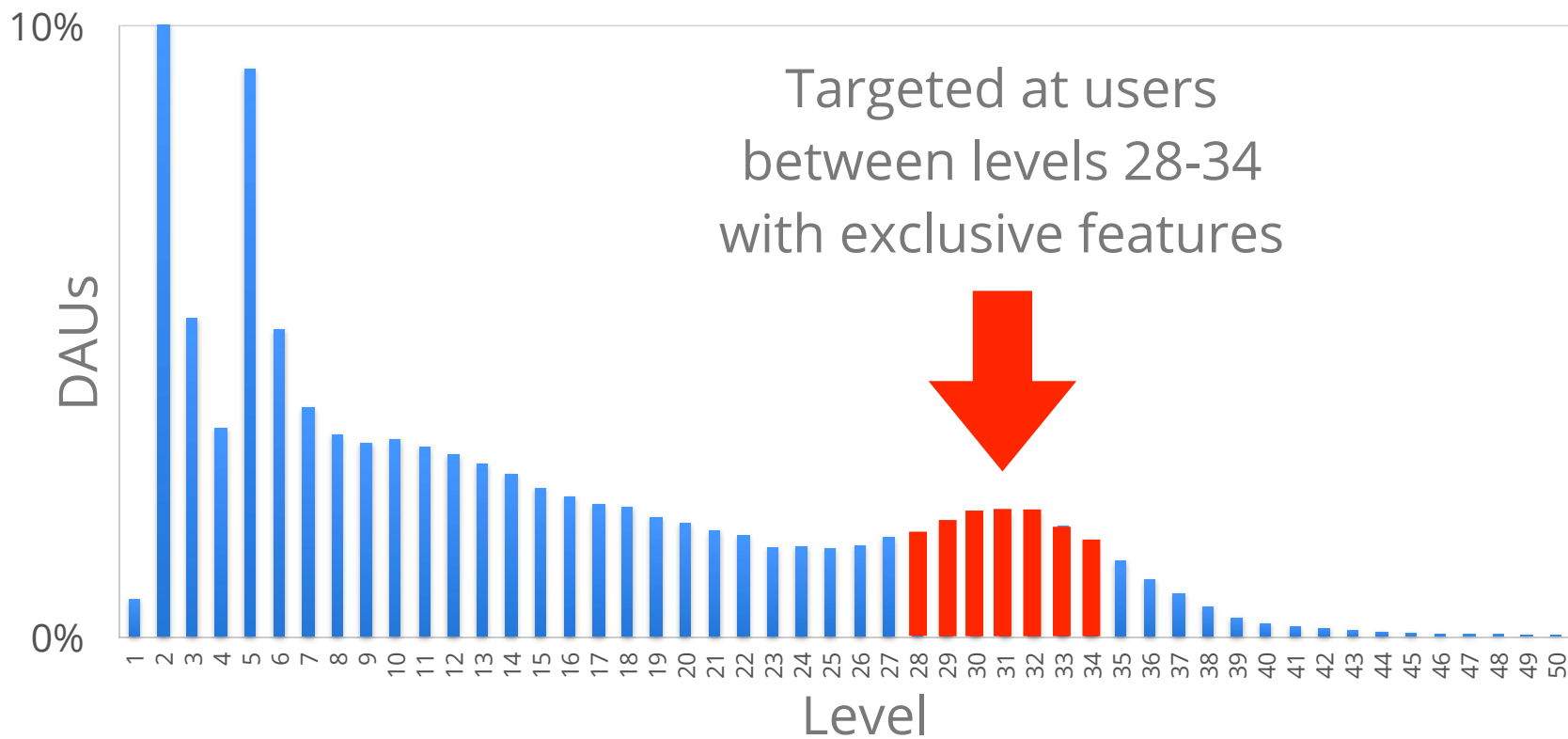
User retention



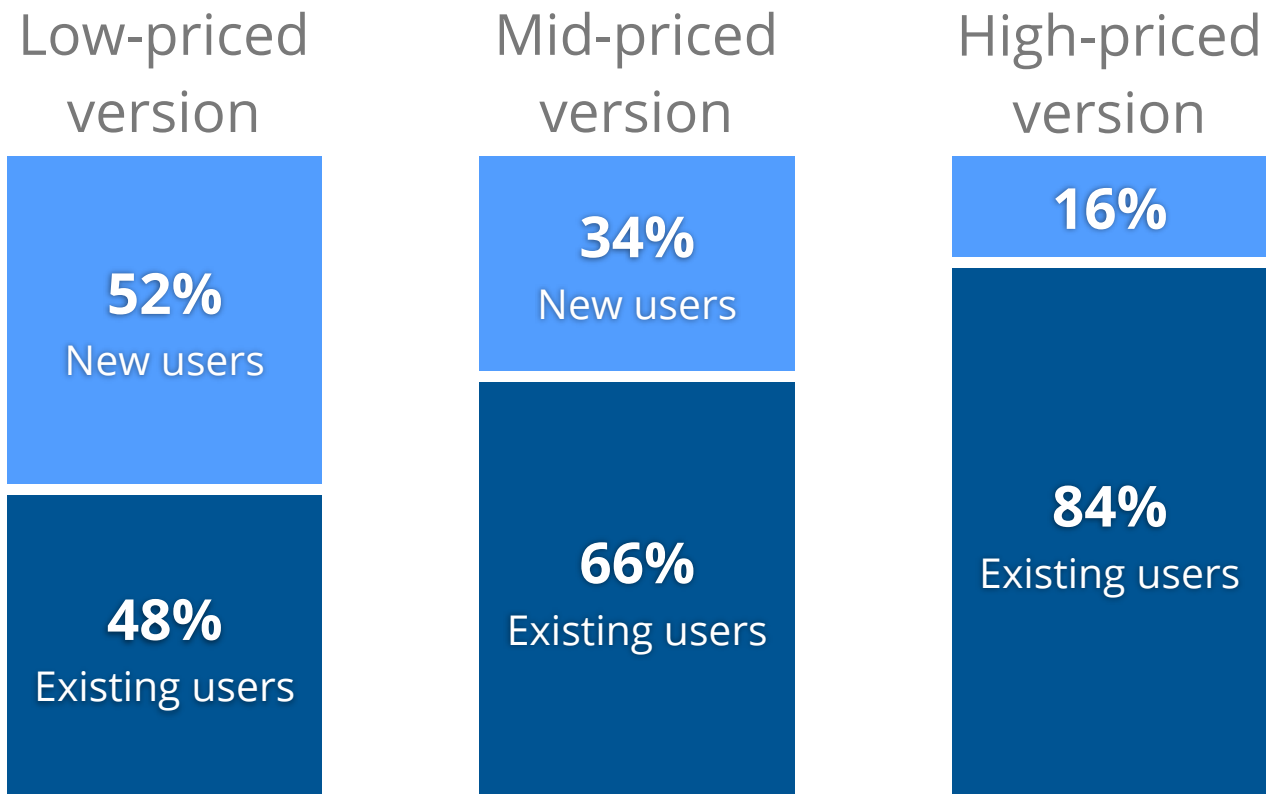
## Tiny Village



# Targeting Users to Increase Engagement



# Targeting Pricing for Conversion





# Targeting with Unique Features to Monetize

Higher level players spend more on items with higher returns

# of Paying Users **1.4X**

Ran a limited-time mini-game with premium features

Avg user spend **1.3X**

Resulted in higher returns and happier players

Revenues **1.9X**



# User Retention

## Launch content frequently



## Social Engagement



## Timely & relevant incentives



# Optimize User Experience

- UI optimized for different devices, screen sizes, aspect ratios
- Adapt to different frame rates
- Native performance



# Monetization Options

**Free:** Ad-supported



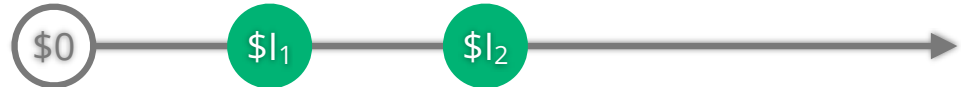
**Premium:** Pay per title



**Trial:** Upgrade to Premium



**Freemium:** In-app items



**Paymium:** Pay per title + In-app items



**Trial Paymium:** Upgrade to Paymium



# Paymium Model - Case Study

## DrawSomething by Zynga

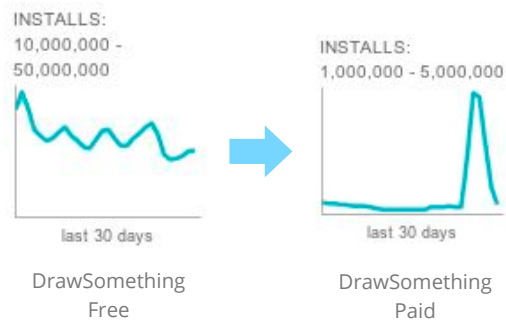
Free version with IAP items; ad-driven

Paid version with IAP items; removes ads

## Distribution

Having 2 separate SKUs  
put the game on 2 lists:  
Top Paid and Top Free

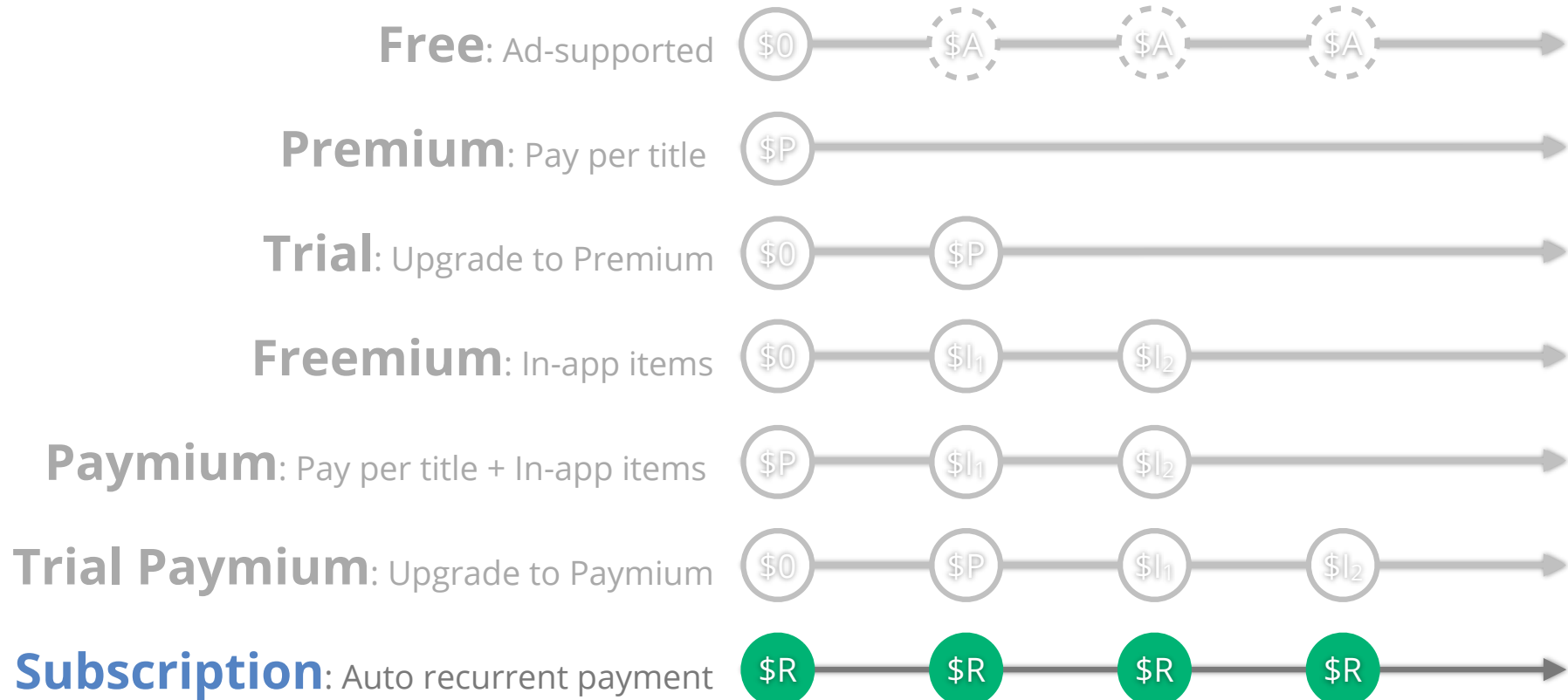
Opportunity to upgrade  
users from free to paid



Effect of flash sale promotion on the Paid version in June

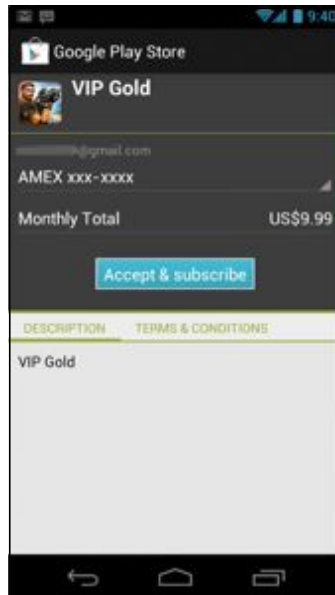


# Options for Monetization

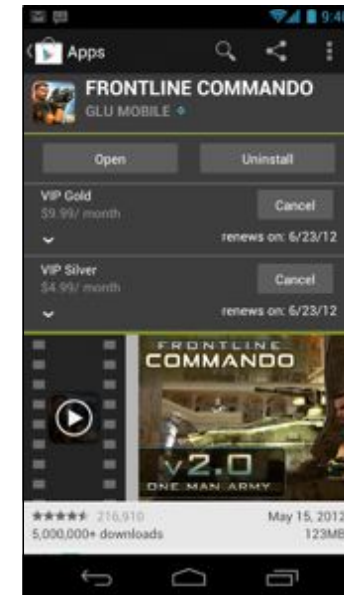


# Subscriptions - Overview

Extension of  
in-app billing



Managed in  
Google Play Store



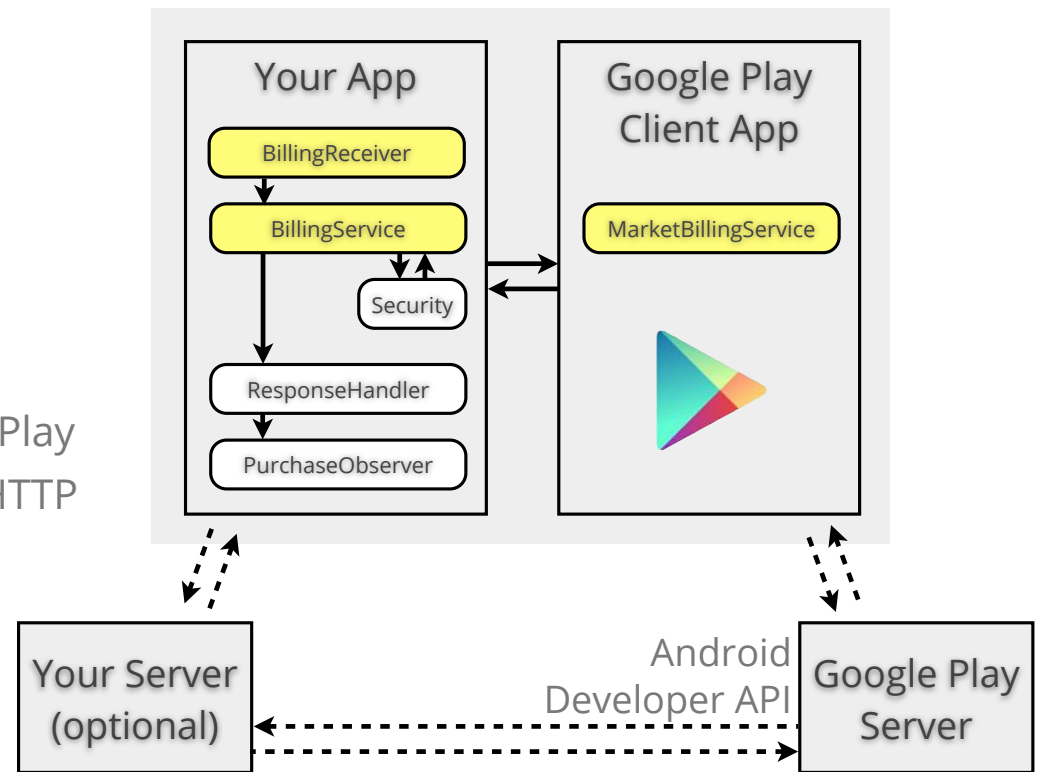
# Subscription Overview

Subscription introduces  
“purchaseToken”

`PURCHASE_STATE_CHANGED`

communicated via

In-app Billing Notifications from Google Play  
Google Play Android Developer API via HTTP

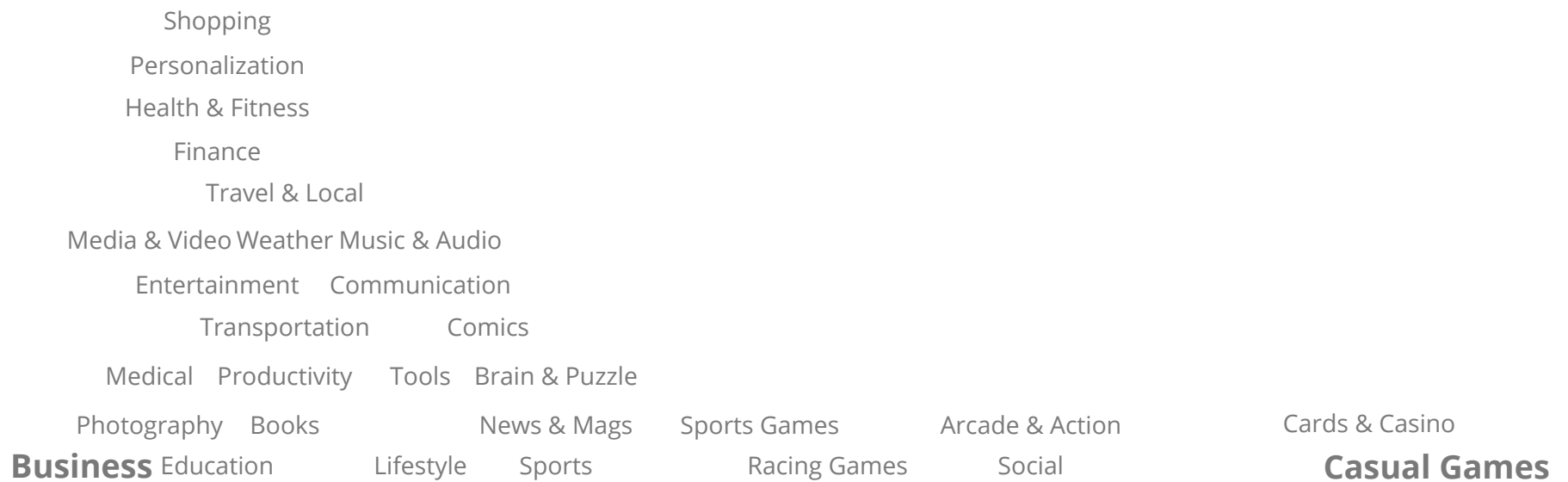




# Monetization Options



# Which Option to Choose?



0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

Number of Freemium Titles in "Top 20 Grossing"



## Key Take Aways

- Find the model and pricing that works for you
- Find the best ways to engage your users
- Launch relevant, targeted content frequently
- Strategy to convert users to paying customers
- Test often to gauge success -> iterate -> repeat
- Optimize user experience across devices
- Social integration and cross-promotions
- Build monetization strategy into app development





# Making money with advertising

# MAD MEN



Image source: AMC Networks



Image source: AMC Networks

i <3 apps

FREE

68%

of apps are free in the  
Android Play store

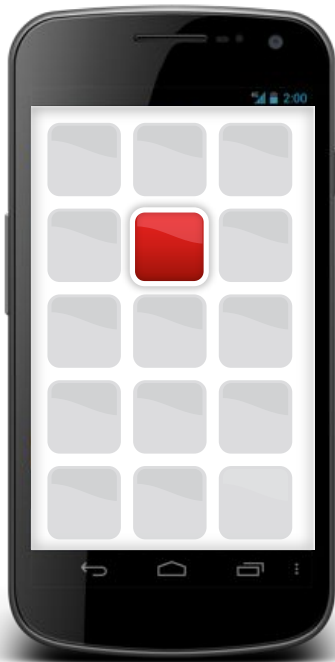


# Mobile app advertising

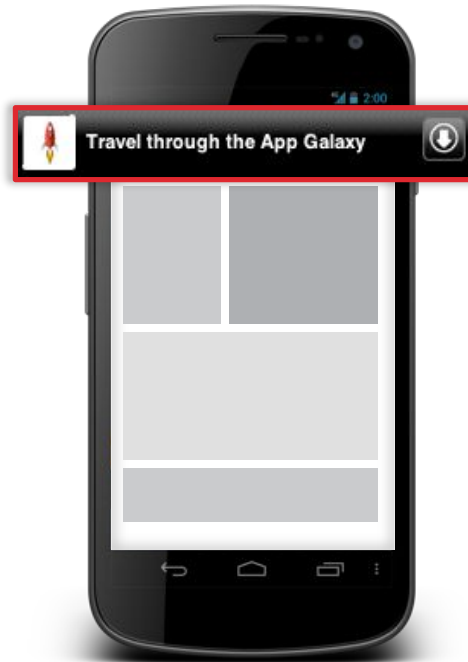




# Building an app business with advertising



User **launches** your app



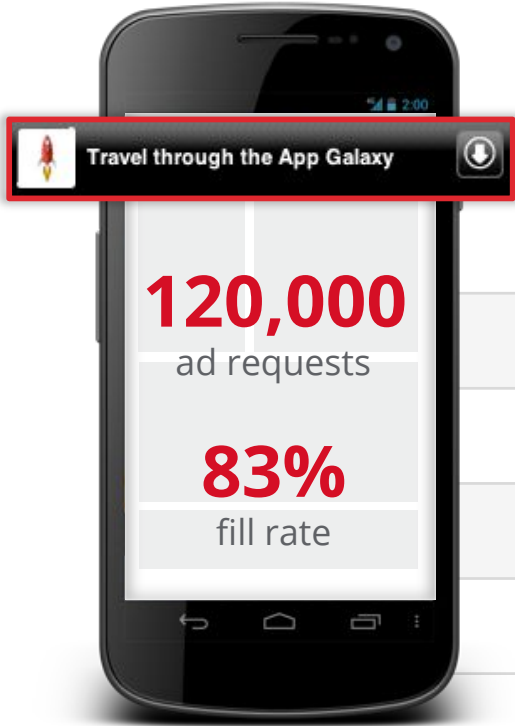
User **clicks** on ad placed in your app



You get **paid** for each click



# How does it all add up?



Impressions		100,000
Click-Through Rate (CTR)		1.5%
<b>Clicks</b>		<b>1,500</b>
Cost Per Click (CPC)		\$0.10
<b>Revenue</b>		<b>\$150</b>



# Adding a mobile banner ad unit

Java

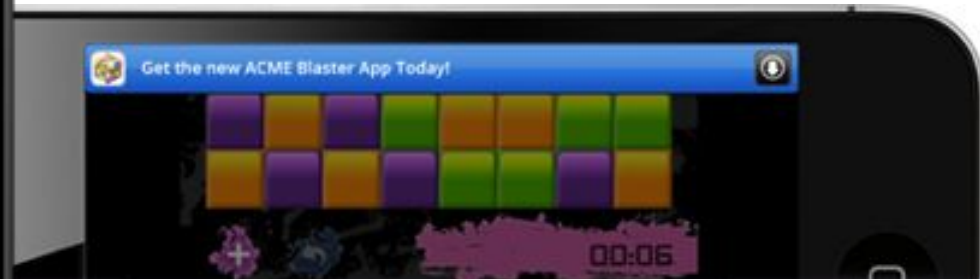
```
public void onCreate(Bundle savedInstanceState) {  
    setContentView(R.layout.main);  
  
    // Create the adView  
    adView = new AdView(this, AdSize.BANNER, "a123456789098765");  
  
    // Lookup your LinearLayout assuming it's been given  
    // the attribute android:id="@+id/mainLayout"  
    LinearLayout layout = (LinearLayout)findViewById(R.id.mainLayout);  
  
    // Add the adView to it  
    layout.addView(adView);  
  
    // Initiate a generic request to load it with an ad  
    adView.loadAd(new AdRequest());  
}
```



## “Smart” banner ads



“Smart” detection of device width and orientation consistently deliver ads that fit



## Ad unit implementation best practices

1. Build with ads in mind



2. Got search?



3. Pay attention to user engagement



4. Avoid accidental clicks



# SimSimi uses AdMob to expand into new markets globally

## Challenge

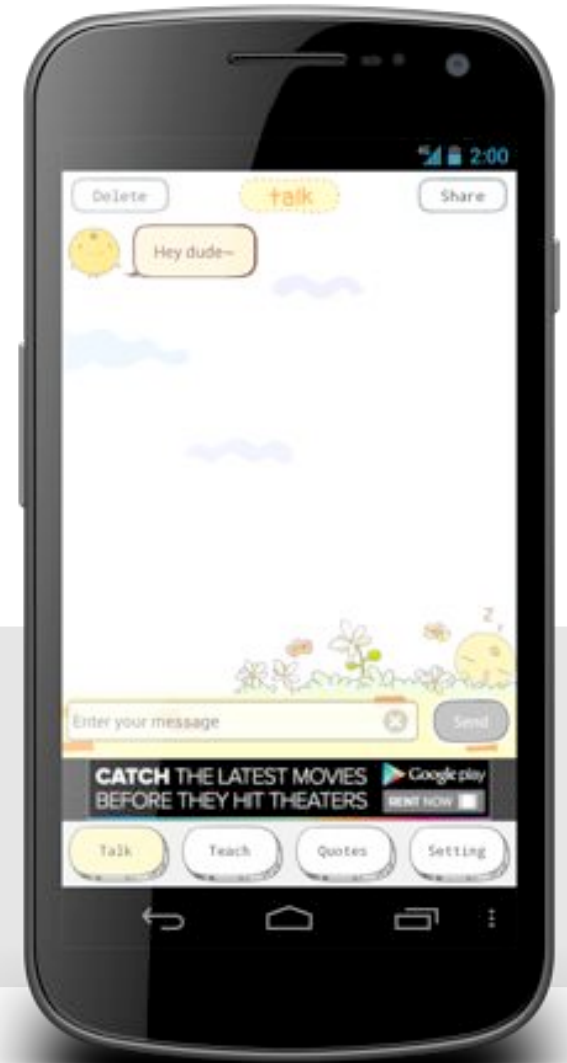
Acquiring paid users globally.

## Approach

Free with AdMob ads for new markets outside of Korea

## Results

15M+ downloads around the world  
80% of downloads from new markets



# SimSimi uses AdMob to expand into new markets globally

“ We chose AdMob because it provided secure revenues as we expanded globally. Our advertising revenues with AdMob soon exceeded our paid app revenues. ”

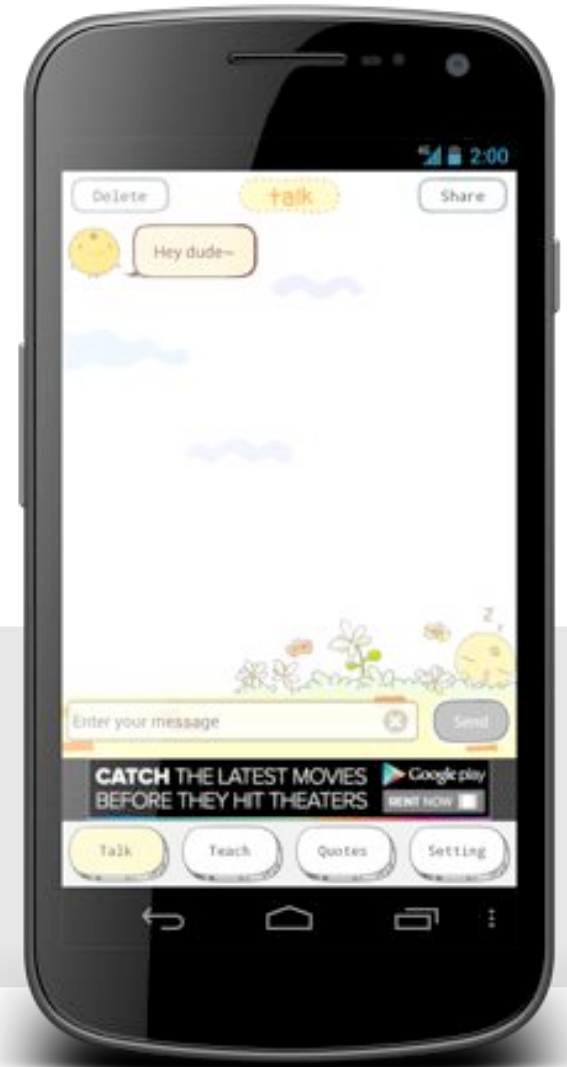
Junghoi Choi, President of SimSimi

## Approach

Free with AdMob ads for new markets outside of Korea

## Results

15M+ downloads around the world  
80% of downloads from new markets



# AroundMe leverages search functionality to grow their app

## Challenge

To provide highly relevant ads that match search results.

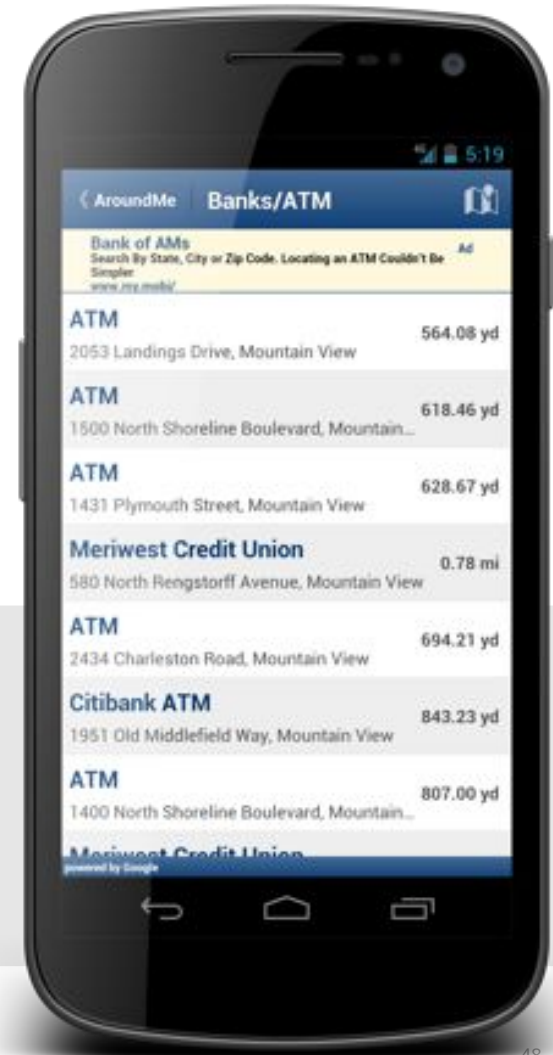
## Approach

Free with Google Search Ads for mobile apps

## Results

3x greater eCPMs

5x higher click-through rates





# AroundMe leverages search functionality to grow their app

“ Custom Search Ads for mobile apps are very useful because we are able to serve ads in real-time that are contextually targeted. ”

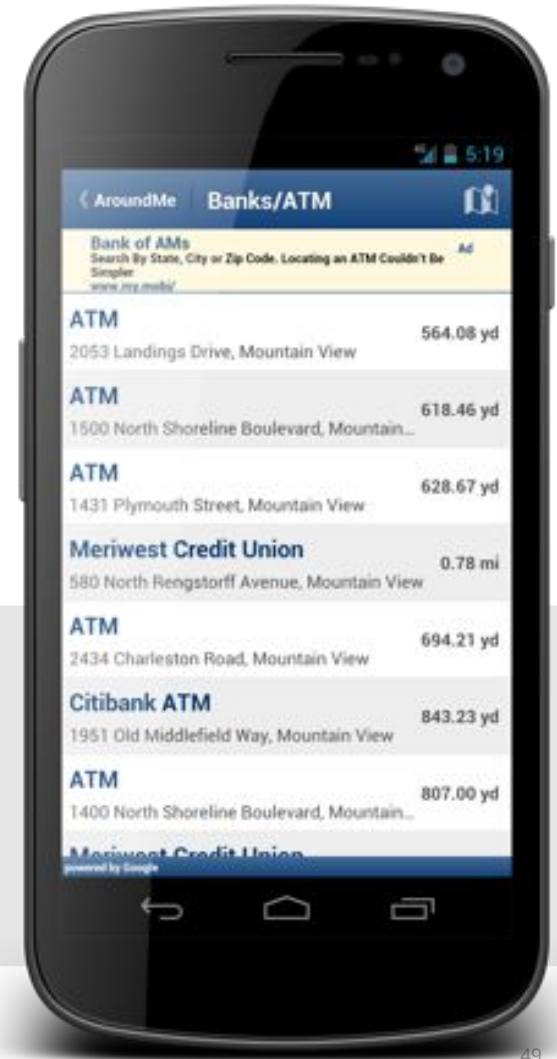
Marco Pifferi, Founder and CEO of AroundMe

## Approach

Free with Google Search Ads for mobile apps

## Results

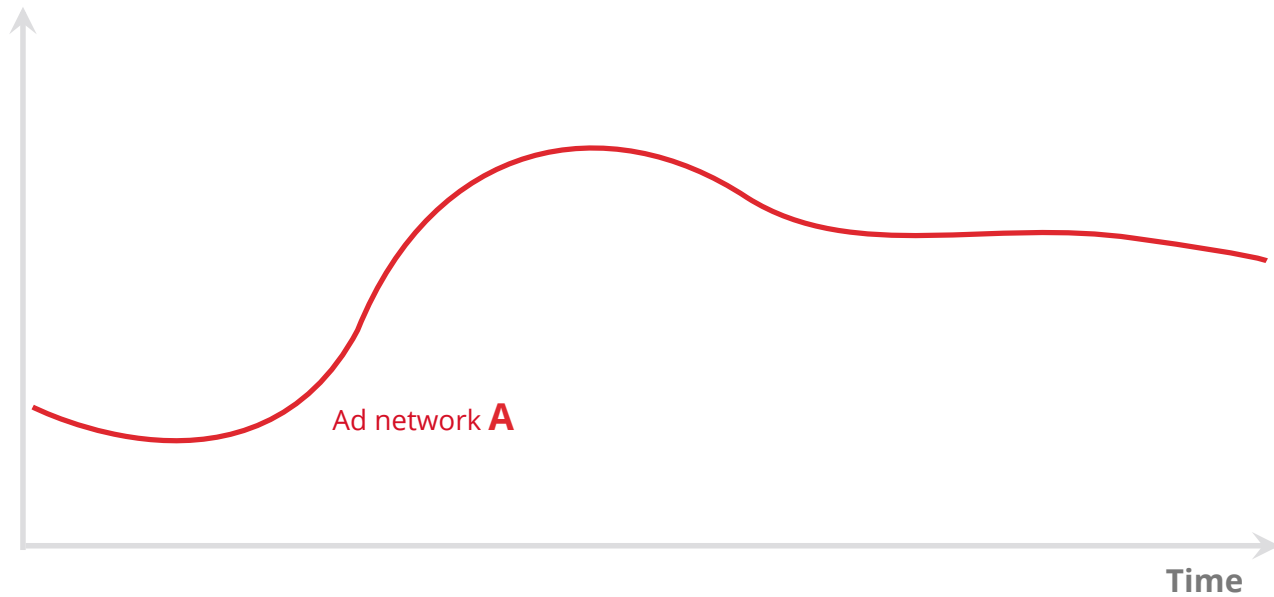
3x greater eCPMs  
5x higher click-through rates



# What about other ad networks?

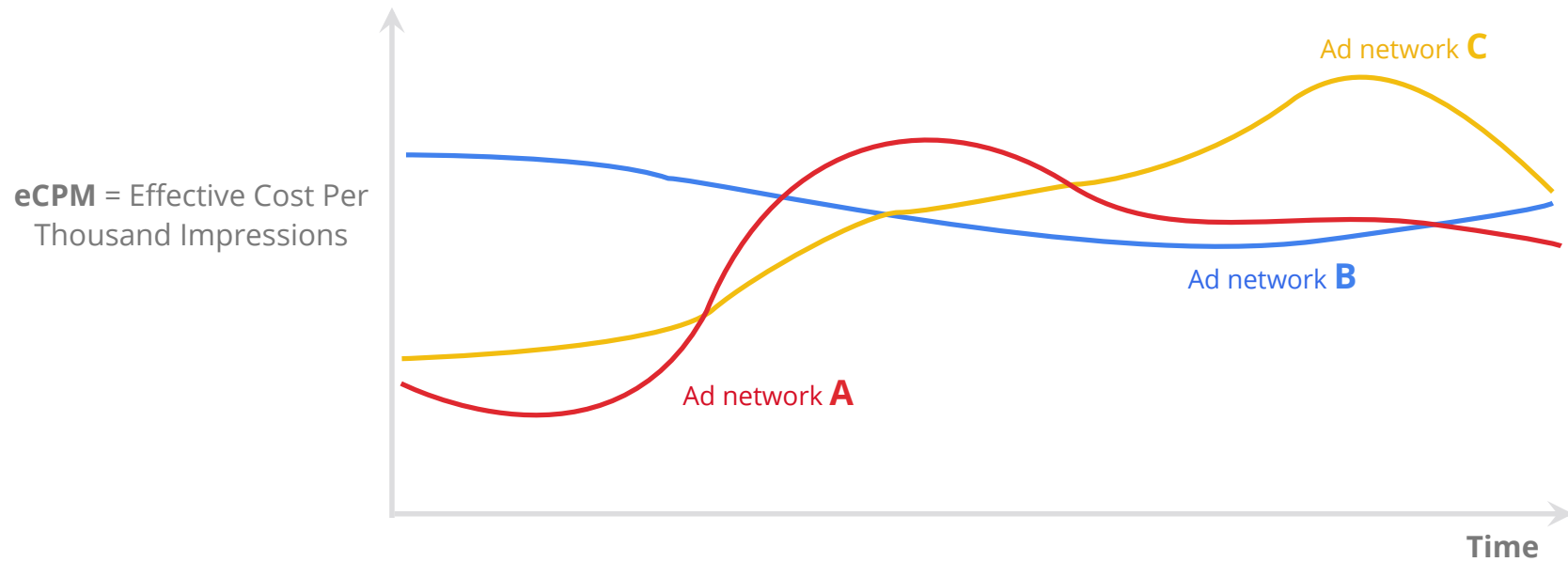
No ad network is created equal

eCPM = Effective Cost Per  
Thousand Impressions



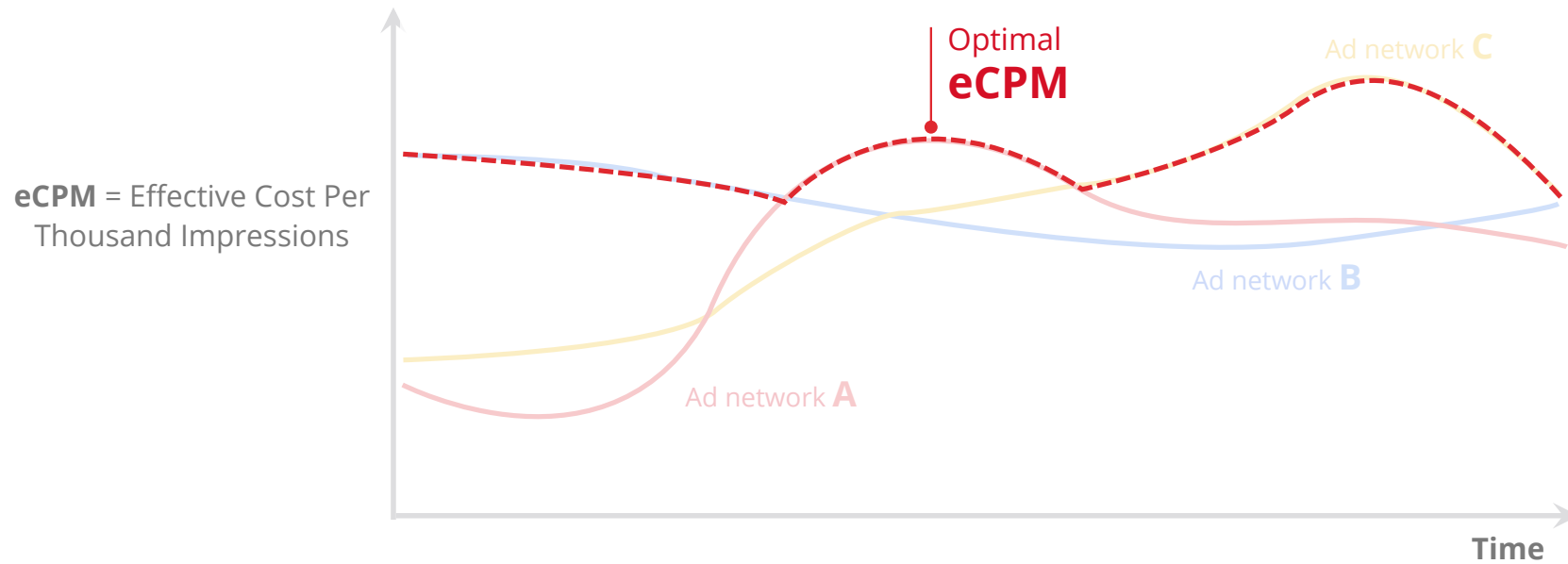
# What about other ad networks?

No ad network is created equal

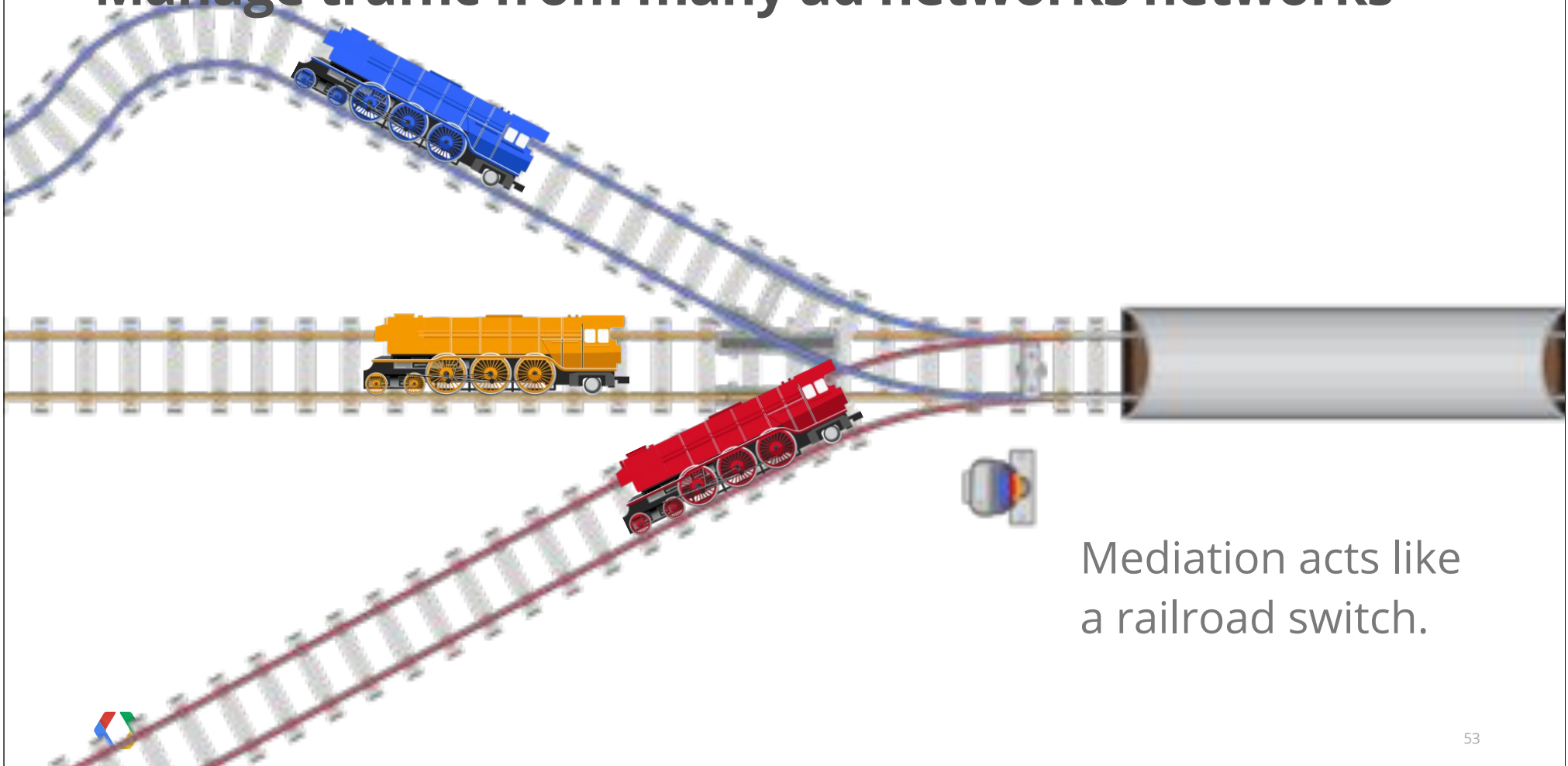


# What about other ad networks?

Mediation helps reduce eCPM fluctuations

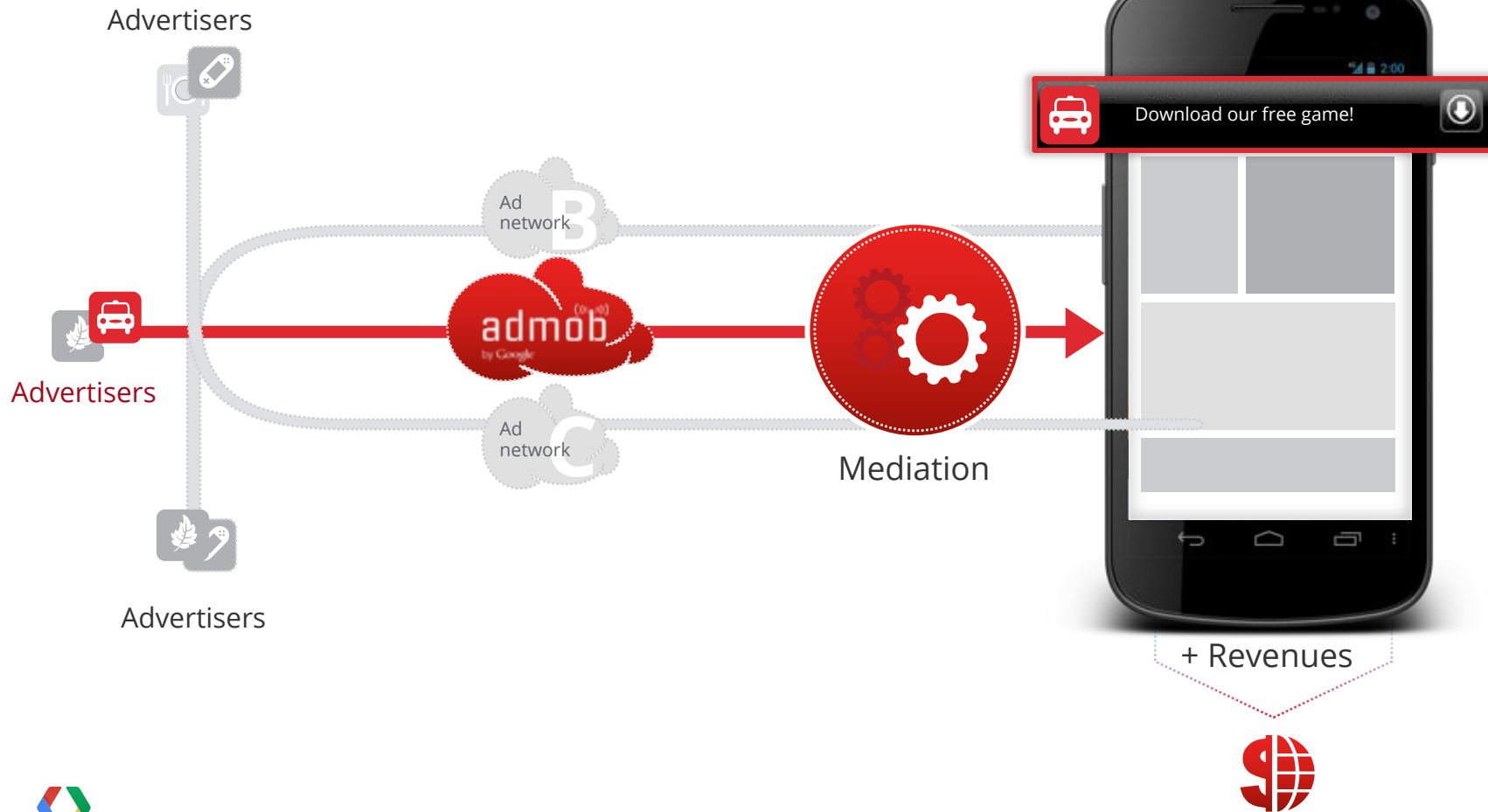


# Manage traffic from many ad networks networks

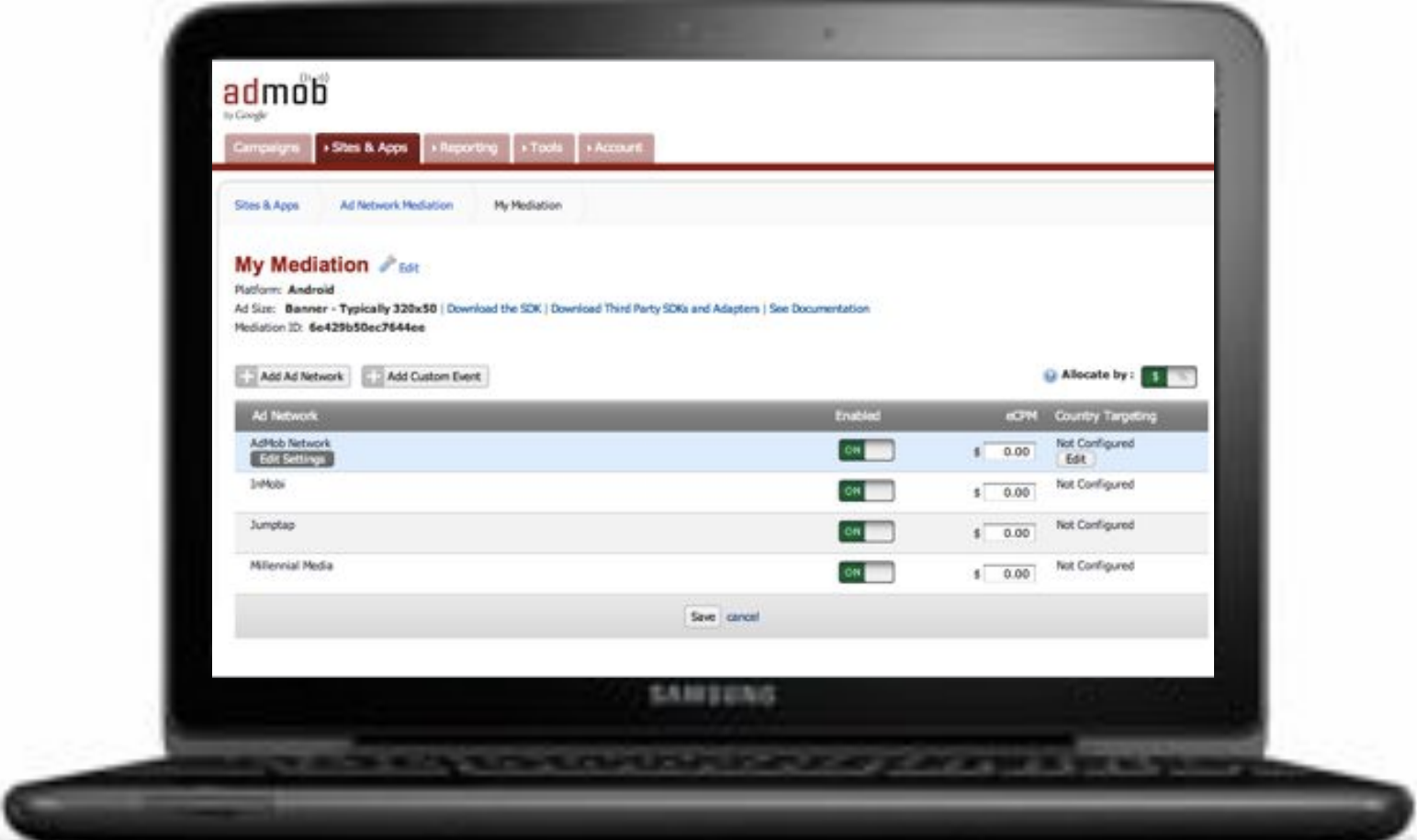


Mediation acts like a railroad switch.

# AdMob ad network mediation



# Many ad networks, one dashboard to manage



# Simply add your mediation ID

Java

```
public void onCreate(Bundle savedInstanceState) {  
    setContentView(R.layout.main);  
  
    // Create the adView  
    adView = new AdView(this, AdSize.BANNER, "a123456789098765");  
  
    // Lookup your LinearLayout assuming it's been given  
    // the attribute android:id="@+id/mainLayout"  
    LinearLayout layout = (LinearLayout)findViewById(R.id.mainLayout);  
  
    // Add the adView to it  
    layout.addView(adView);  
  
    // Initiate a generic request to load it with an ad  
    adView.loadAd(new AdRequest());  
}
```





# Simply add your mediation ID

Java

```
public void onCreate(Bundle savedInstanceState) {
    setContentView(R.layout.main);

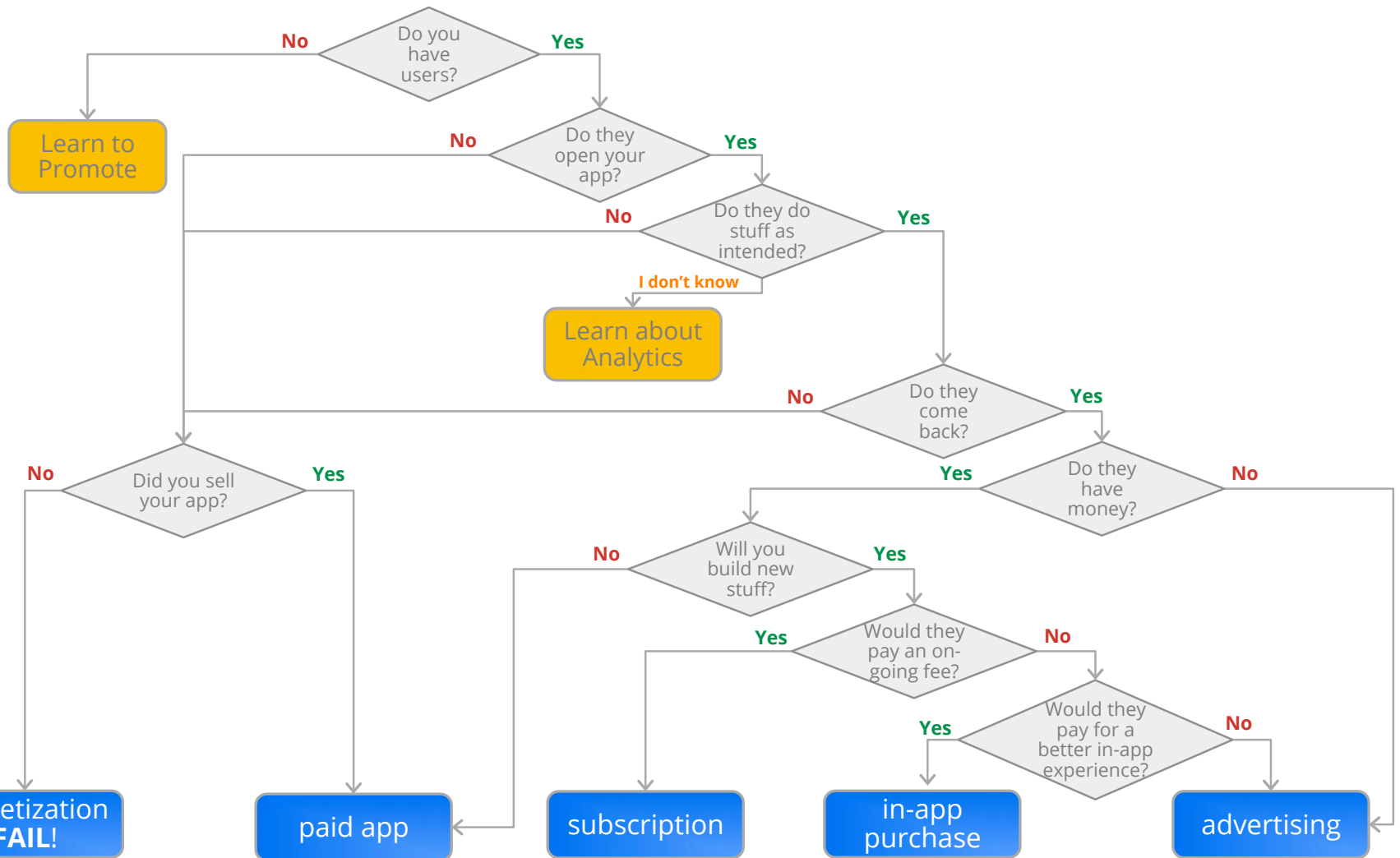
    // Create the adView
    adView = new AdView(this, AdSize.BANNER, "e987654321012345");

    // Lookup your LinearLayout assuming it's been given
    // the attribute android:id="@+id/mainLayout"
    LinearLayout layout = (LinearLayout)findViewById(R.id.mainLayout);

    // Add the adView to it
    layout.addView(adView);

    // Initiate a generic request to load it with an ad
    adView.loadAd(new AdRequest());
}
```





<Thank You!>

[play.google.com](https://play.google.com)  
[admob.com](https://admob.com)





Google  
Developers