



Making Good Apps Great

More advanced techniques for expert Android developers

Reto Meier

Android Developer Relations Tech Lead



Android Protips 2

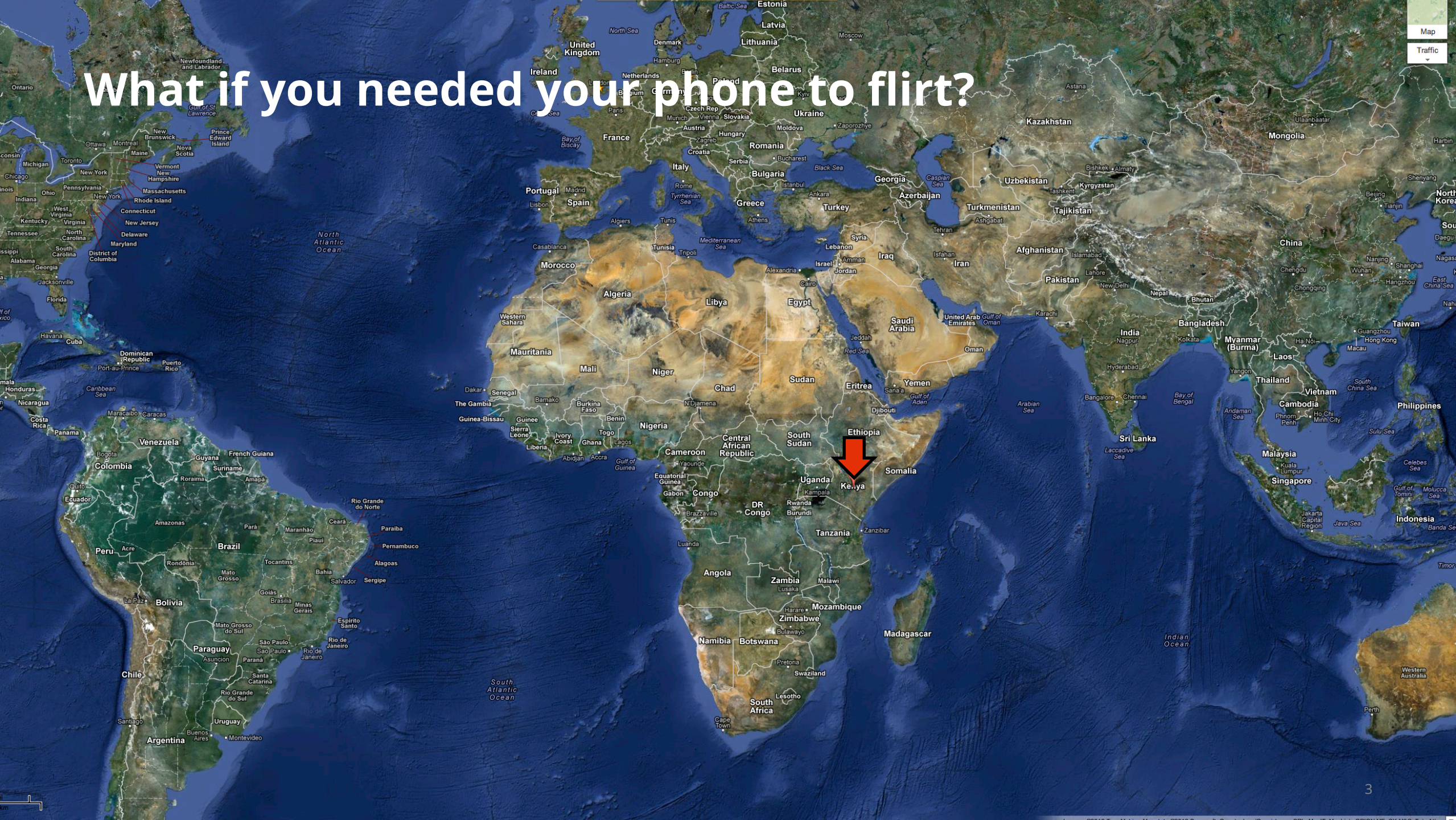
Electric Boogaloo

Reto Meier

Android Developer Relations Tech Lead



What if you needed your phone to flirt?



“I don't use the Internet on my phone...”

Smart phone user



“I don’t use the Internet on my phone.
Of course I use Google and Gmail.”

Smart phone user



When Does the Internet Stop Being the Internet?



Invisibility



Efficiency



Reliability



ACHIEVEMENT!
INVISIBILITY



LIFE 

Apps You Don't Think About

- Work offline
- Be consistent but creative
- Know that less is more
- Understand not all devices are the same



Queue and Send

MyIntentService.java

```
if (!isConnected) {
    alarms.cancel(retryQueuedCheckinsPendingIntent);
    pm.setComponentEnabledSetting(connectivityReceiver,
        PackageManager.COMPONENT_ENABLED_STATE_ENABLED, PackageManager.DONT_KILL_APP);
    addToQueue(timeStamp, reference, id);
} else {
    if (!checkin(timeStamp, reference, id))
        addToQueue(timeStamp, reference, id);
    // Retry each of the queued checkins
    // Delete the queued checkins that were successful.
    // If there are still queued checkins then set a non-waking alarm to retry them.
    if (queuedCheckins.getCount() > 0) {
        long trigger = SystemClock.elapsedRealtime() + RETRY_INTERVAL;
        alarms.set(AlarmManager.ELAPSED_REALTIME, trigger, retryQueuedCheckinsPendingIntent);
    }
}
```



Apps You Don't Think About

- Work offline
- Be consistent but creative
- Know that less is more
- Understand not all devices are the same



Android Design

[*http://developer.android.com/design*](http://developer.android.com/design)

GET STARTED ^

Creative Vision

Design Principles

UI Overview

STYLE v

PATTERNS v

BUILDING BLOCKS v

DOWNLOADS

DEVELOPERS



Welcome to **Android Design**, your place for learning how to design exceptional Android apps.

Creative Vision >

Apps You Don't Think About

- Work offline
- Be consistent but creative
- Know that *less is more*
- Understand not all devices are the same



Apps You Don't Think About

- Work offline
- Be consistent but creative
- Know that less is more
- Understand not all devices are the same



Different Devices, Different Hardware



Detect Your Hardware Platform

MyService.java

```
PackageManager pm = getPackageManager();
```

```
boolean hasBFCamera = pm.hasSystemFeature(PackageManager.FEATURE_CAMERA);
```

```
boolean hasWiFiDirect = pm.hasSystemFeature(PackageManager.FEATURE_WIFI_DIRECT);
```

```
boolean hasPortScreen = pm.hasSystemFeature(PackageManager.FEATURE_SCREEN_PORTRAIT);
```



I/O 


SCORE 0006521887

MOUNTAIN VIEW
GOOGLE → 6 KM

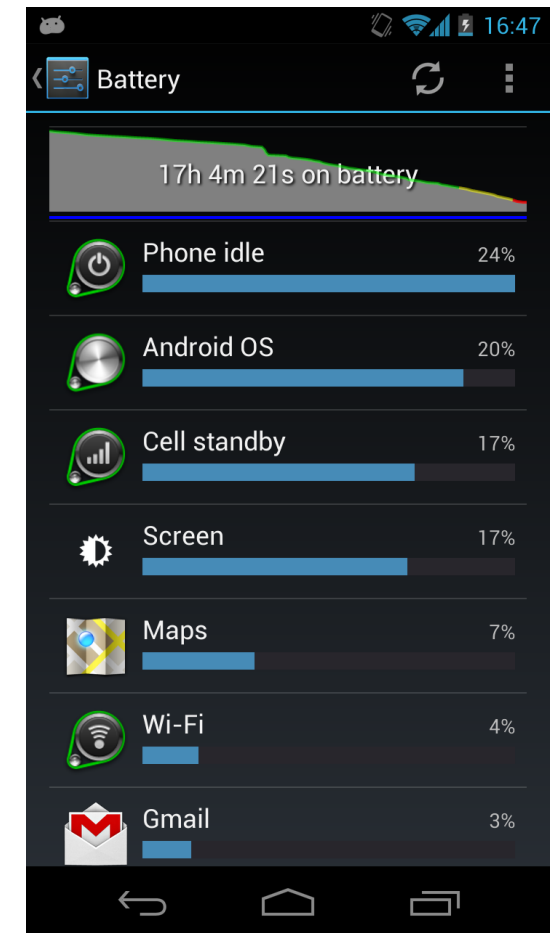
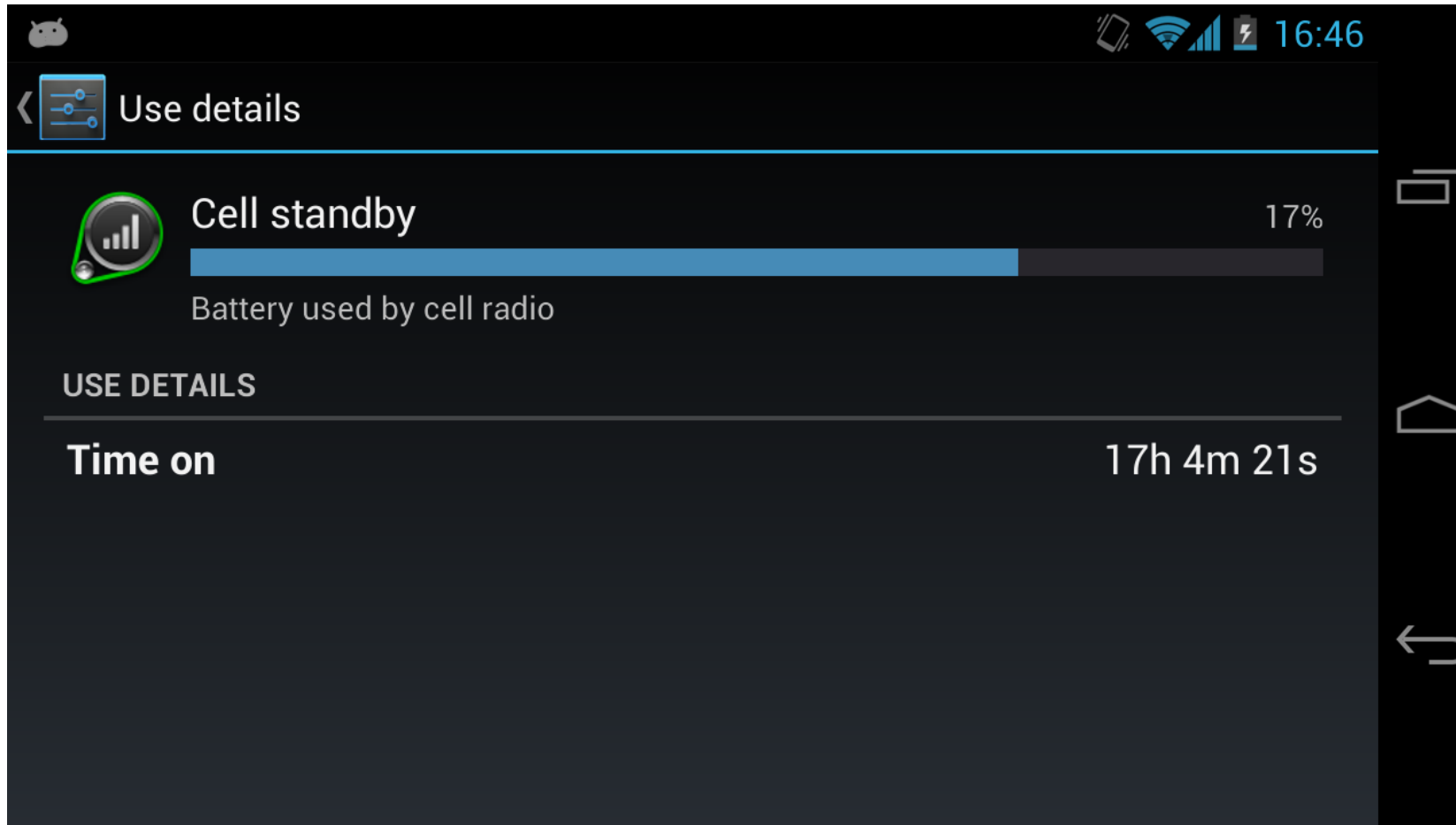
EXIT ↓

ACHIEVEMENT!
EFFICIENCY!

ANDROID
EXIT



Boost Battery Life by Enabling Airplane Mode



Using the mobile radio less

- Smaller payloads
- Transfer less often
- Cache your results





**Fewer large
downloads?**



**Many small
downloads?**



The False Economy of the Little Cookie

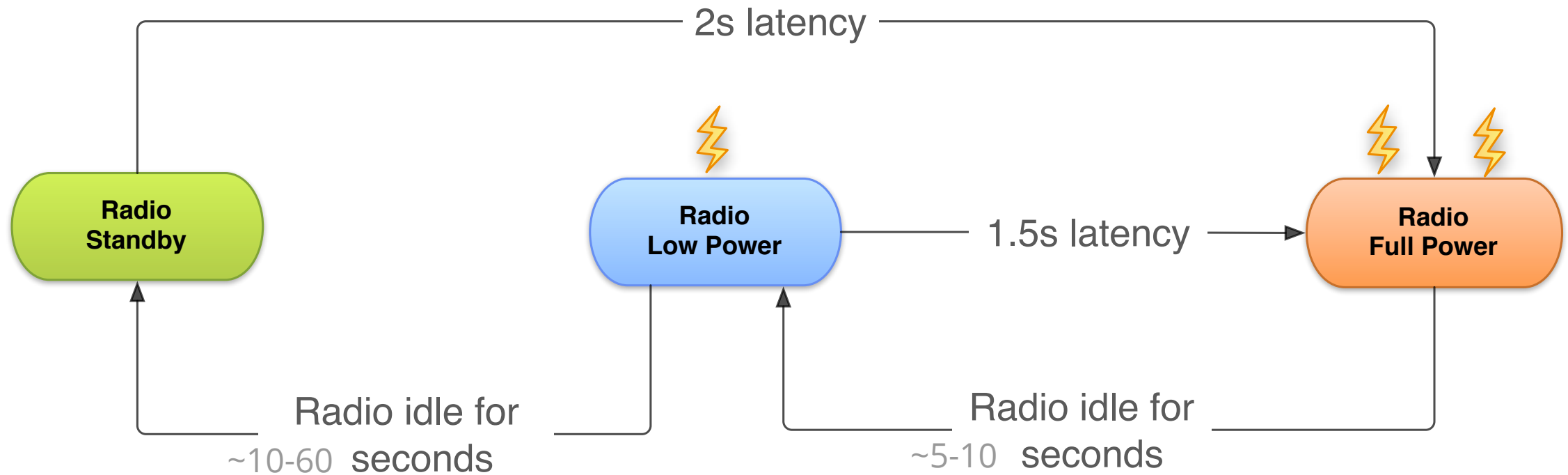
- Transfer less data across the network
- Store and process less data on the device
- Use less **memory** / storage / **bandwidth**



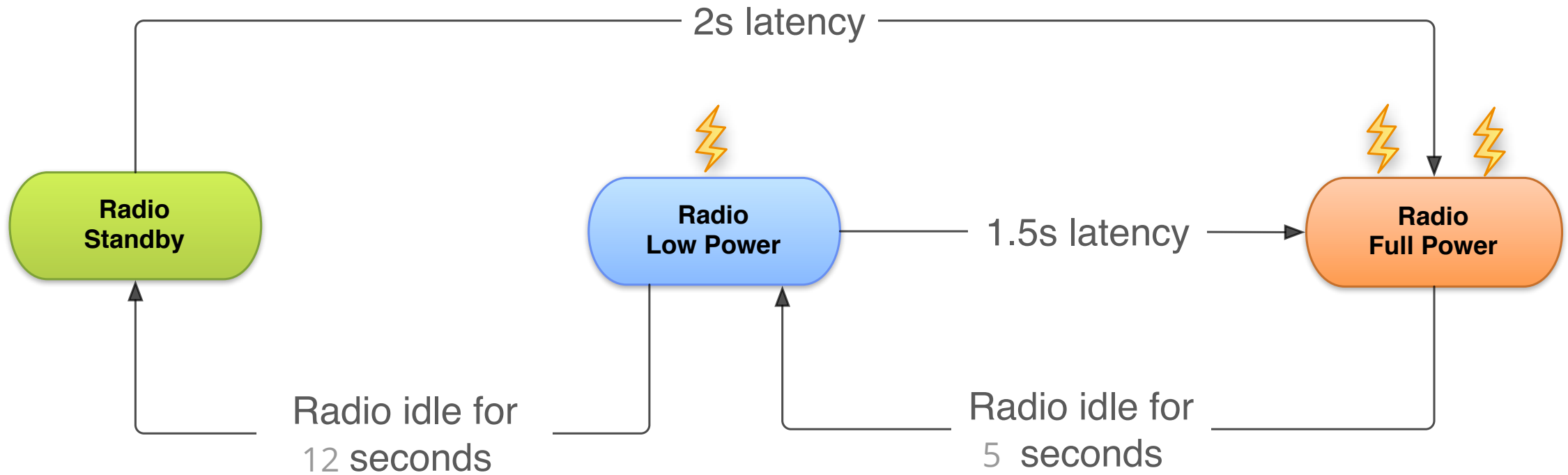
The Big Cookie Model



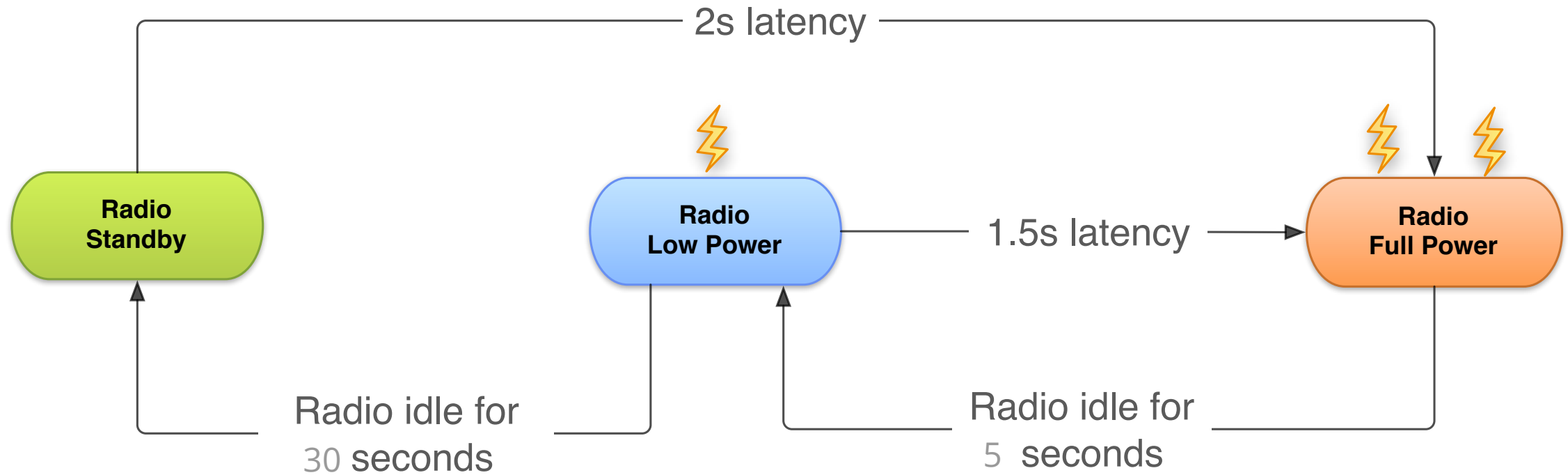
The Mobile Radio State Machine



AT&T 3G



Vodafone



Normalizing Mean State Transition Times with FFTs

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} x(k) e^{-jk2\pi n/N}$$

where N = number of carriers

\forall carrier \in (supported carriers)

$$n = \frac{\sqrt{\alpha^2 \times \beta x^2 \times \gamma^2}}{\pi}$$

where

$$\alpha = \Delta(DCH \rightarrow FACH)$$

$$\beta = \Delta(FACH \rightarrow IDLE)$$

$$\gamma = (\Delta(IDLE \rightarrow DCH) / \Delta(FACH \rightarrow DCH)) / 2$$

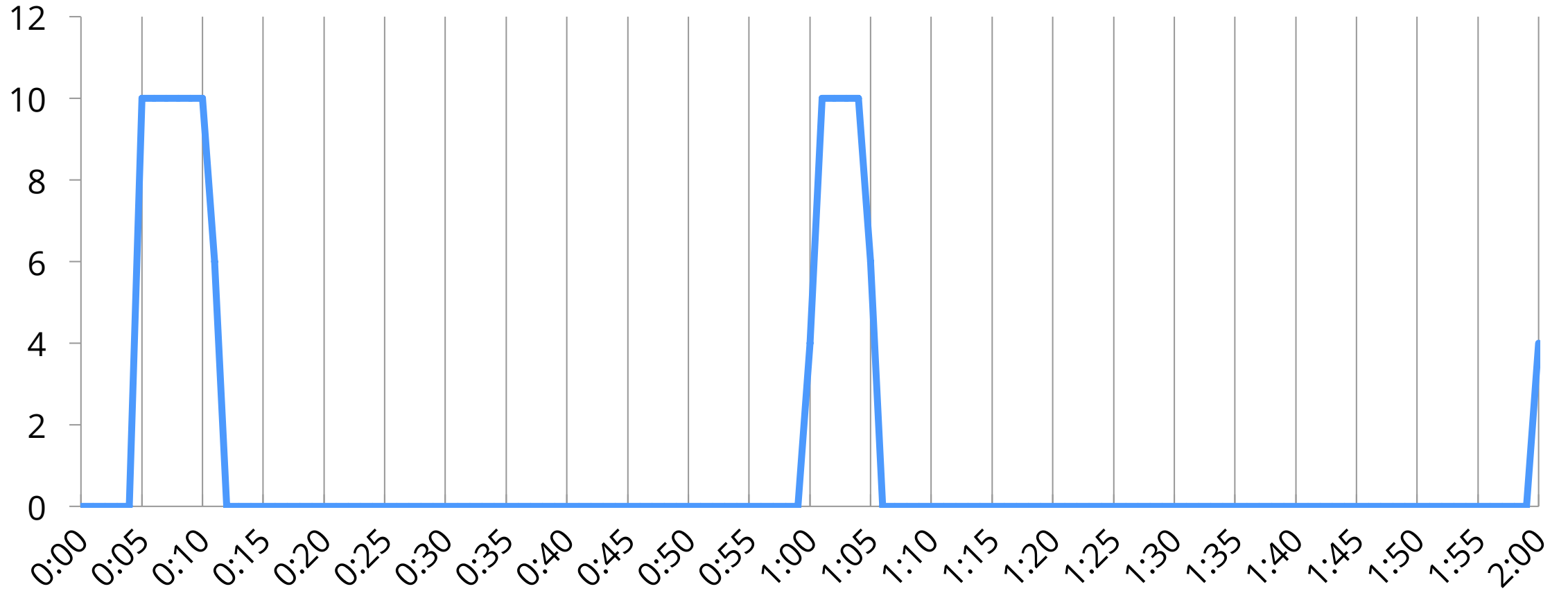


Defragmenting Your Network Traffic

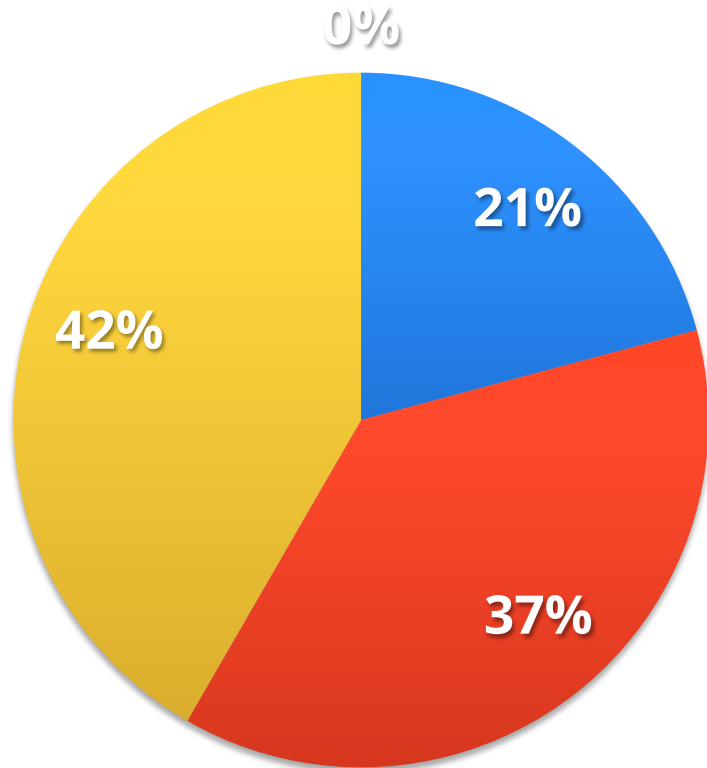
- Prefetching
- Batching, bundling, and preempting
- Reducing your number of connections



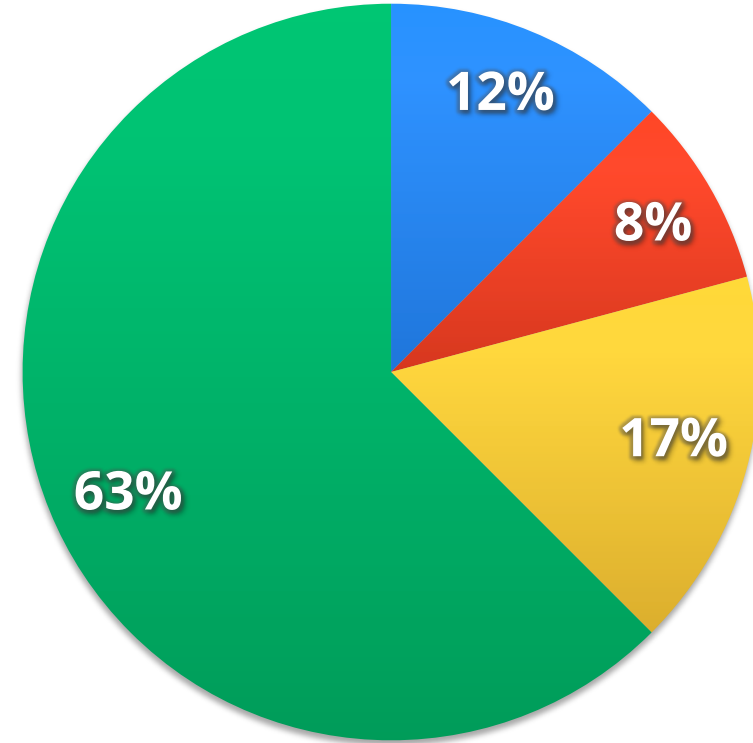
Defragmented Network Traffic



Same Data. Different Power Profile.

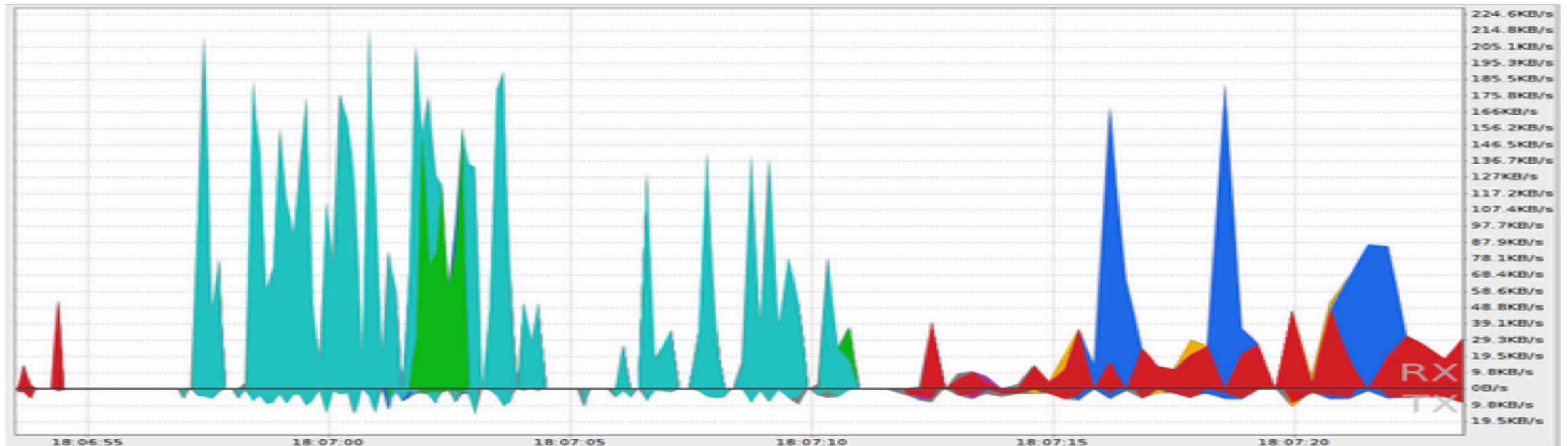


- Active
- High Power
- Low Power
- Idle



Analyzing Network Usage

- Logcat logging
- ARO tool from AT&T
- Network Statistics in DDMS



Tag your Data Transfers

MyActivity.java

```
TrafficStats.setThreadStatsTag(0xF00D);  
TrafficStats.tagSocket(outputSocket);  
  
// TODO Transfer data using socket  
  
TrafficStats.untagSocket(outputSocket);
```



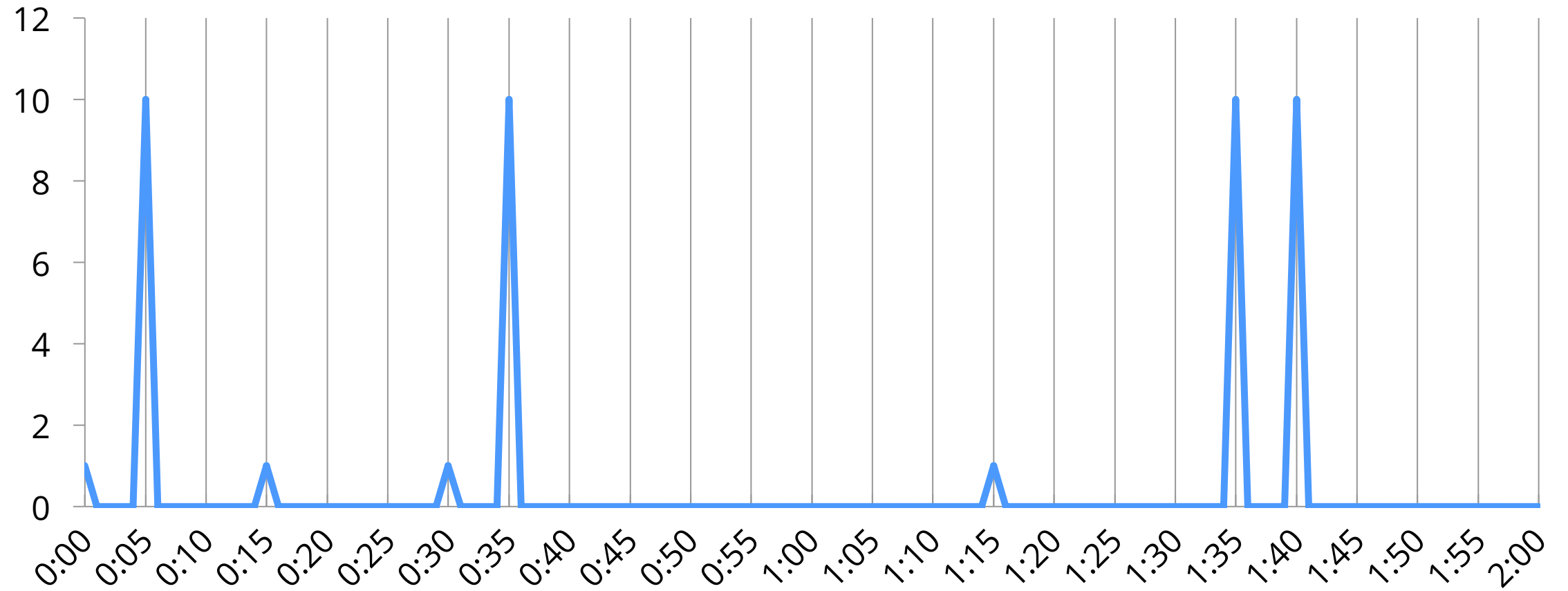
Tag your Data Transfers

MyActivity.java

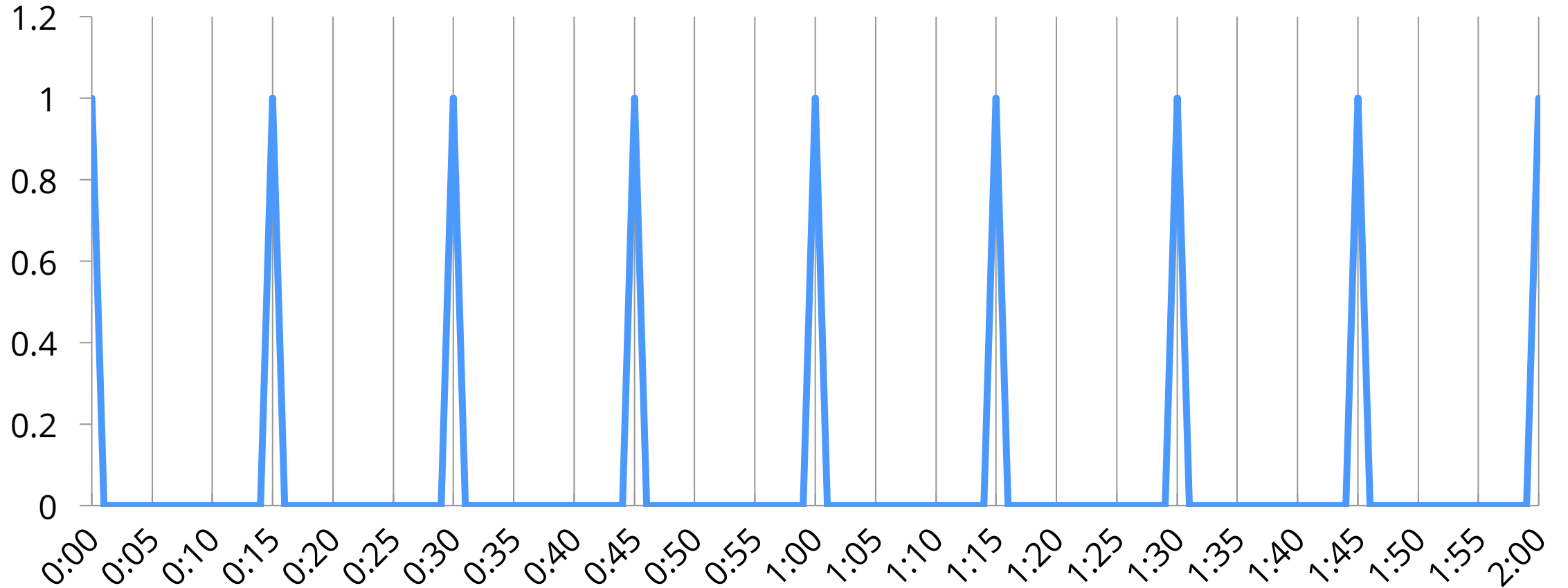
```
TrafficStats.setThreadStatsTag(0xF00D);  
try {  
    // Make network request using HttpClient.execute()  
} finally {  
    TrafficStats.clearThreadStatsTag();  
}
```



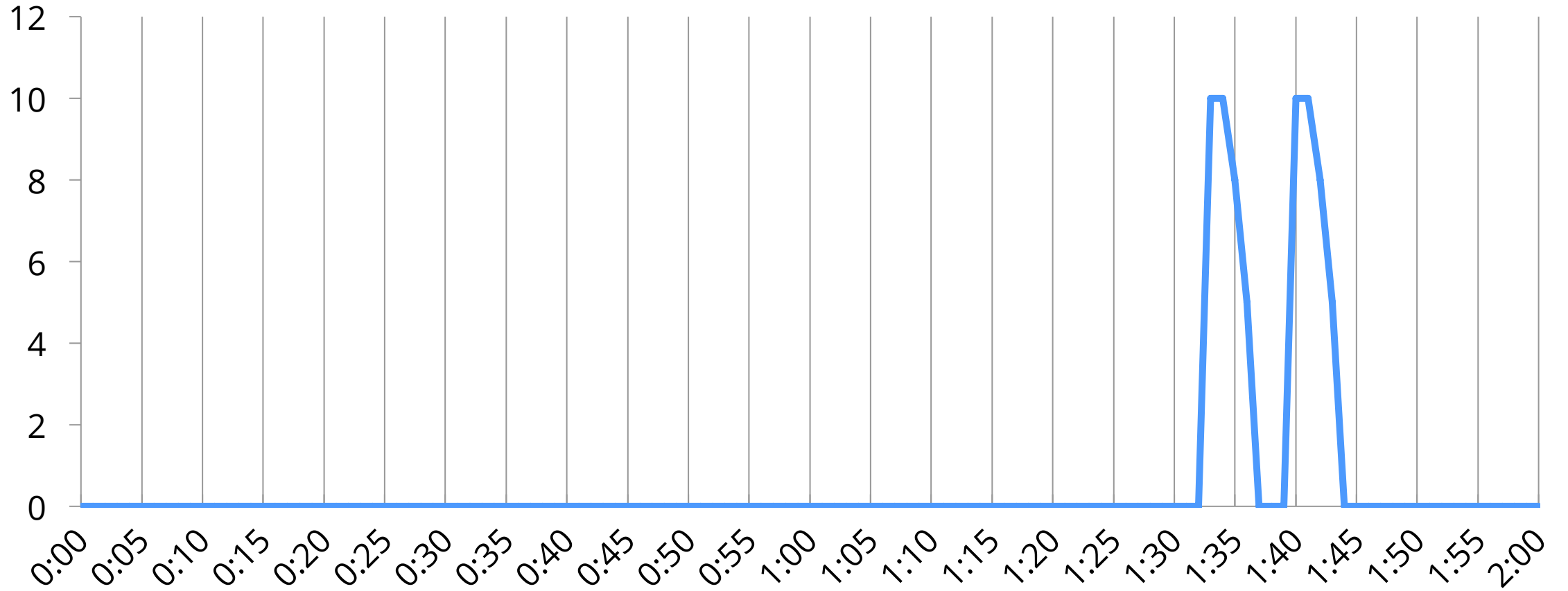
Short Spikes



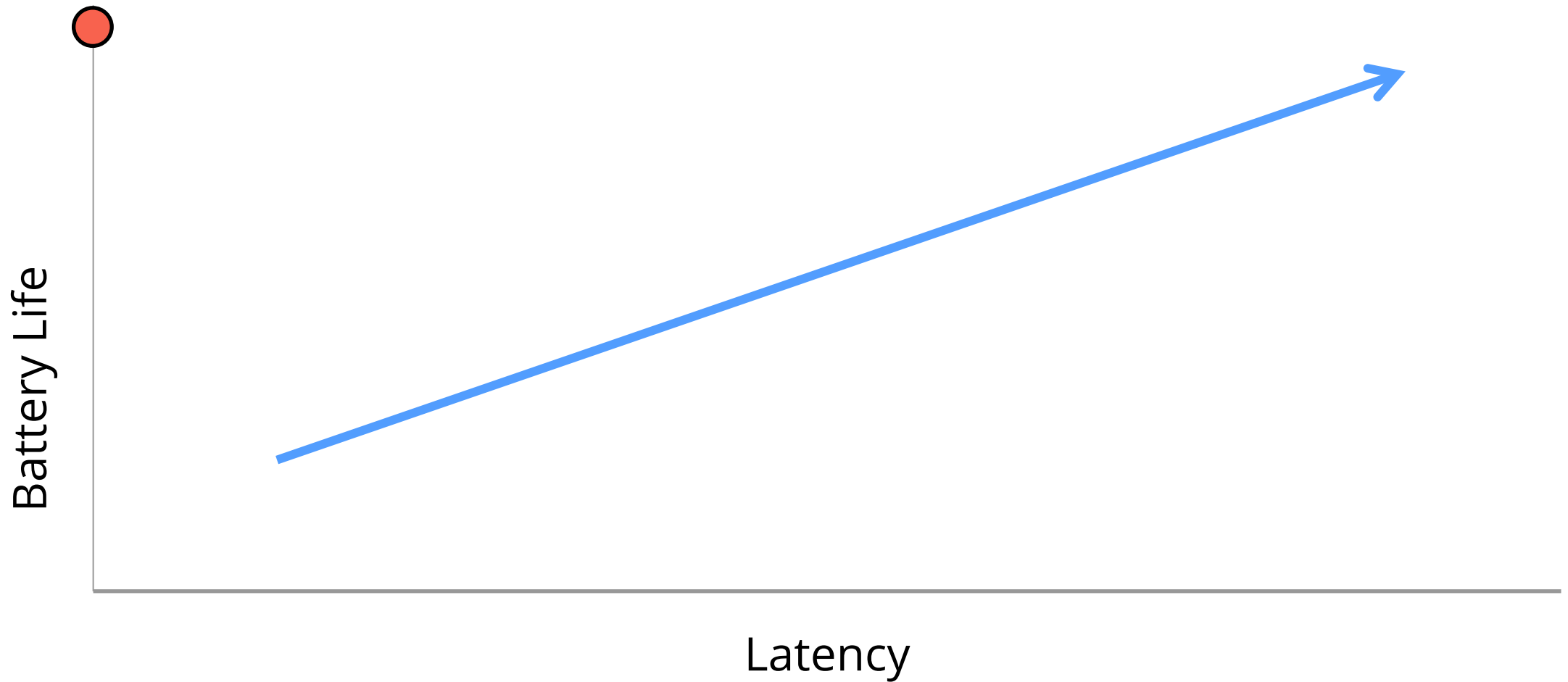
Regular / Periodic Transfers



Batches of Activity in Close Proximity



Battery Life versus Latency



“A good app is like a good butler.
It has what you want, before
you have to ask for it.”



Prefetching

- Try to download only what you'll need
- 2 to 5 minutes of app usage
- 1 to 5mb of data (on a 3G network)



Not All Networks Transfer Data Equally

MyService.java

```
int prefetchCacheSize = DEFAULT_PREFETCH_CACHE;
switch (activeNetwork.getType()) {
    case ConnectivityManager.TYPE_WIFI:
        prefetchCacheSize = MAX_PREFETCH_CACHE; break;
    case ConnectivityManager.TYPE_MOBILE): {
        switch (telephonyManager.getNetworkType()) {
            case TelephonyManager.NETWORK_TYPE_LTE:
            case TelephonyManager.NETWORK_TYPE_HSPAP:
                prefetchCacheSize *= 4; break;
            case TelephonyManager.NETWORK_TYPE_EDGE:
            case TelephonyManager.NETWORK_TYPE_GPRS:
                prefetchCacheSize /= 2; break;
            default: break;
        } break;
    }
    default: break;
}
```



Batching and Preempting

- Transfer as much as possible during each session
- Minimize the number of sessions
- Delay time-insensitive transfers
- Preempt scheduled transfers



Create a Batch Queue for Periodic Transfers

TransferQueueSingleton.java

```
private Queue<MyPeriodicTransfer> updateQueue;

public synchronized void enqueuePeriodicTransfer(MyPeriodicTransfer periodicTransfer) {
    updateQueue.add(periodicTransfer);
}

public void executeBatchedPeriodicTransfers() {
    // Execute the batched periodic update queue.
    executeBatchedPeriodicTransfersOnly();
    // Preempt scheduled update
    executeNextPrefetch();
}

private synchronized void executeBatchedPeriodicTransfersOnly() {
    // TODO Bundle the received updates / requests into a single transfer.
    updateQueue.clear();
    // TODO Upload / download the periodic transfer
}
```



Trigger Pending Transfers During On Demand Updates

TransferQueueSingleton.java

```
public void executeOnDemandDownload(DownloadDetails details) {  
    // TODO Execute an on demand download.  
    executeNextPrefetch();  
}
```

```
public void executeNextPrefetch() {  
    // TODO Execute the next planned prefetch.  
  
    // Execute the batched periodic update queue  
    executeBatchedPeriodicTransfersOnly();  
}
```





Minimize the Impact of Regular Transfers

Inexact Repeating Alarms

MyService.java

```
int alarmType = AlarmManager.ELAPSED_REALTIME;  
long interval = AlarmManager.INTERVAL_HOUR;  
long start = SystemClock.elapsedRealtime() + interval;  
  
alarmManager.setInexactRepeating(alarmType, start, interval, pi);
```



Inactivity Back-off

executePeriodicUpdate()

```
boolean appUsed = prefs.getBoolean(PREF_APPUSED, false);
long updateInterval = prefs.getLong(PREF_INTERVAL, DEFAULT_REFRESH_INTERVAL);

if (!appUsed)
    if ((updateInterval * 2) > MAX_REFRESH_INTERVAL)
        updateInterval = MAX_REFRESH_INTERVAL;

reschedulePeriodicUpdates(updateInterval); // Save interval & reschedule alarm.
executeUpdate();                          // Execute data transfer.
```



Failure Back-off

SJ Media Systems
Pro Audio & Video FOR LESS
Warehouse PRICES!
AdChoices
SJ MEDIA SYSTEMS
SHOP NOW

ESPN cricinfo
Welcome Guest
Sign In | Register | Log In
Search
Popular: Kevin Pietersen | ICC rankings | James Anderson

Series Countries Live Scores Fixtures Results News Features Photos Video & Audio Blogs Statistics Archive Games Mobile

England v West Indies home | News | Features | Photos | Fixtures | Results | Squads | Grounds | Tour Stats | Records | History

England v West Indies on **ESPN cricinfo** is in association with **Emirates**

Page Not Found

The page you requested was not found

</england-v-west-indies-2012/engine/match/5342111.html>

This might be because:

- You may have typed the web address incorrectly. Please check the address and spelling ensuring that it does not contain spaces.
- An incorrect link was given either on an internal or external page.
- It is possible that the page you were looking for may have been moved, updated or deleted.
- Please make sure you have refreshed/reloaded the page you were on before this one. There are many changes to the links on our website every day.

Please try one of the following:

- Go to our list of currently scored **live matches**
- View our list of **scorecards of recently completed matches**
- Go to the main **Cricinfo.com** homepage
- Use the search box on the left of the screen

If you think you have found a broken link please let us know:
Your name

Emirates
Hello Tomorrow
Fly Emirates to 21 destinations in Africa.
Book now

cricinfo widgets
CRICKET NEWS, LIVE SCORES & STATISTICS

404 = 131kb

*60

*24

*7

= 1320480kb

= 1.26 GIGABYTES



Failure Back-off



404. That's an error.

The requested URL /1231 was not found on this server.
That's all we know.



$$\begin{aligned} 404 &= 4\text{kb} \\ &*60 \\ &*24 \\ &*7 \\ &= 40320\text{kb} \\ &= 39\text{mb} \end{aligned}$$



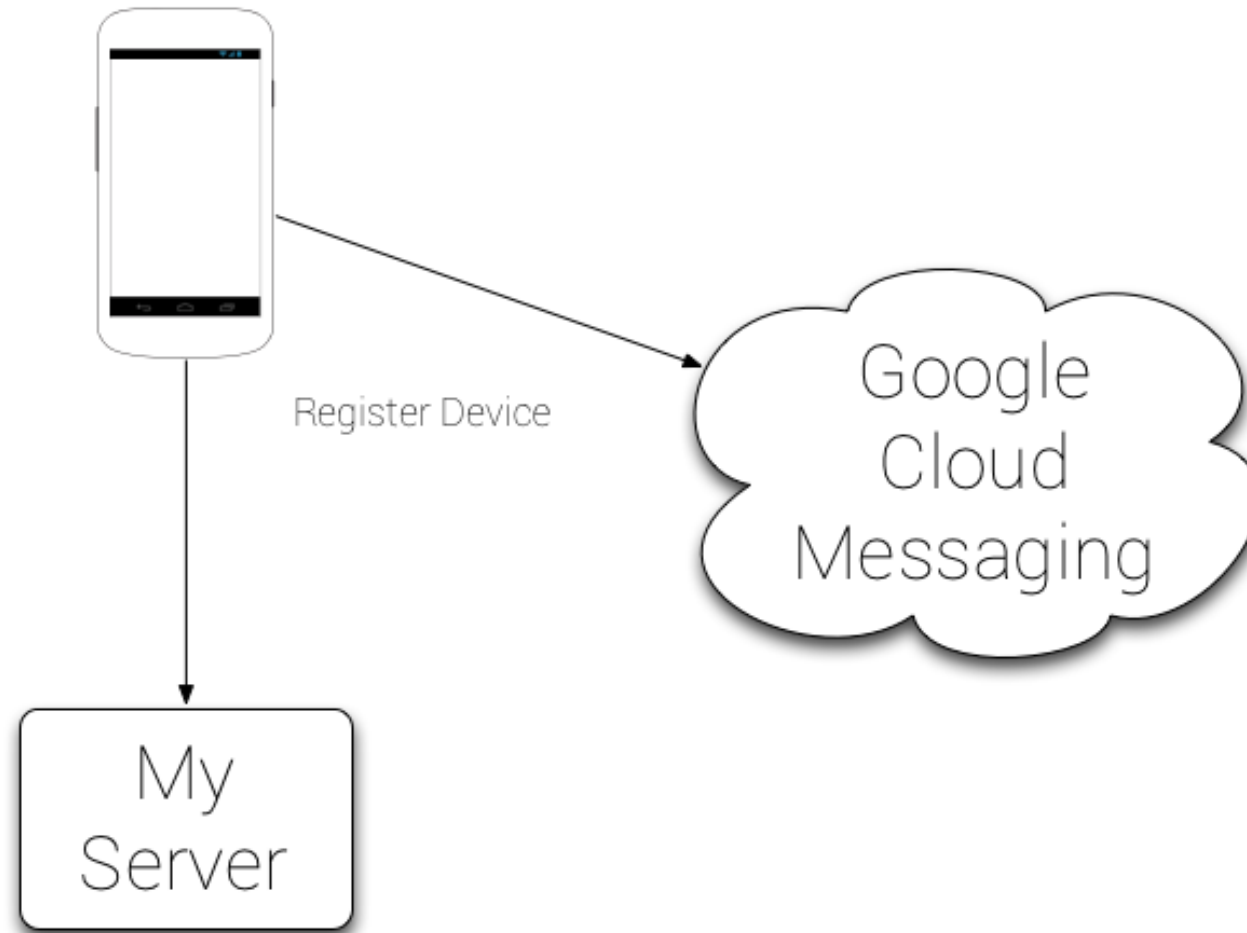
Failure Back-off

MyService.java

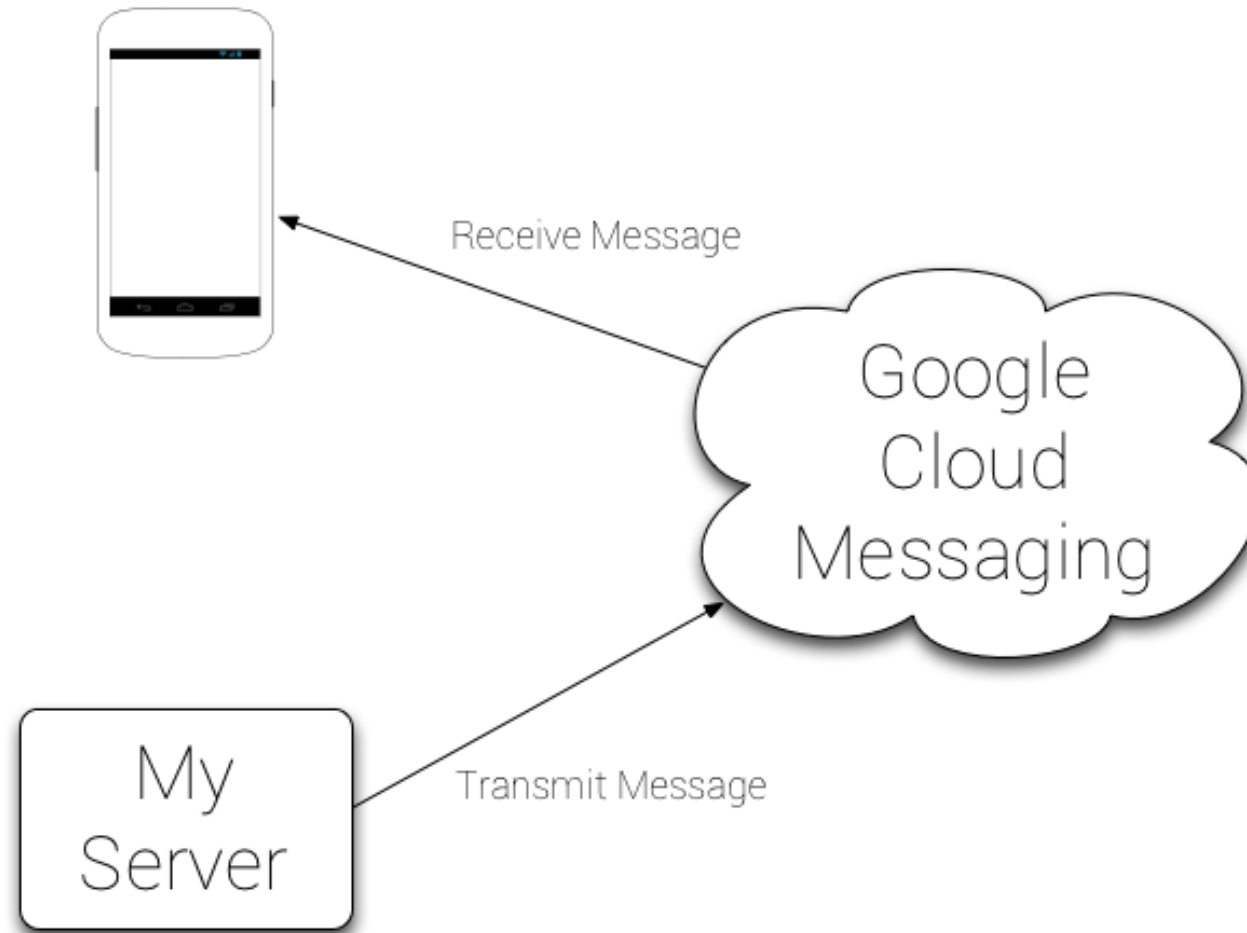
```
private void retryIn(long interval) {  
    Thread.sleep(interval);  
    boolean success = attemptTransfer();  
  
    if (!success) {  
        retryIn(interval*2 < MAX_RETRY_INTERVAL ?  
            interval*2 : MAX_RETRY_INTERVAL);  
    }  
}
```



Google Cloud Messaging



Google Cloud Messaging



Google Cloud Messaging

- Available back to Android 2.2 (~94% of devices)
- Can transmit to multiple recipients
- No quota limits
- Tickle not transmit
- Next session in [Room 9](#): **Google Cloud Messaging**





**Redundant Downloads are
Redundant**

Reduce Your Payloads

- Filter on the server
- Rescale your images on the server
- Cache ALL the things!

MyService.java

```
// Non-sensitive data
Context.getExternalCacheDir();

// Sandboxed application data
Context.getCacheDir();
```



Don't Download Again Until Necessary

MyService.java

```
long expires = httpURLConnection.getHeaderFieldDate("Expires", currentTime);
long lastModified = httpURLConnection.getHeaderFieldDate("Last-Modified", currentTime);

// Don't refresh until at least the expiry time
setDataExpirationDate(expires);

if (lastModified > lastUpdateTime) {
    // Parse update
}
```



Use the HttpResponseCache

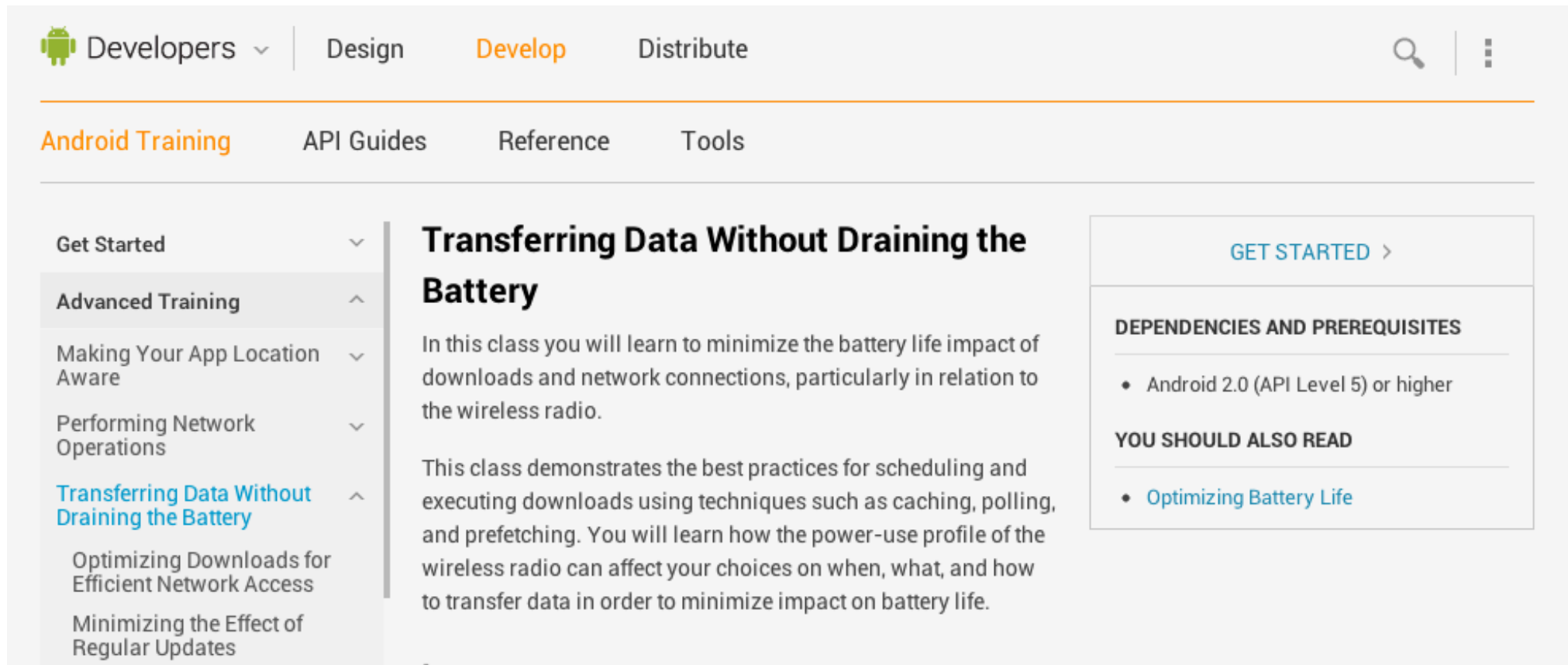
MyService.java

```
private void enableHttpResponseCache() {  
    try {  
        long httpCacheSize = 10 * 1024 * 1024; // 10 MiB  
        File httpCacheDir = new File(getCacheDir(), "http");  
        Class.forName("android.net.http.HttpResponseCache")  
            .getMethod("install", File.class, long.class)  
            .invoke(null, httpCacheDir, httpCacheSize);  
    } catch (Exception httpResponseCacheNotAvailable) {  
        Log.d(TAG, "HTTP response cache is unavailable.");  
    }  
}
```



Transferring Data Without Draining the Battery

<http://developer.android.com/training/efficient-downloads>



The screenshot shows the Android Developer website interface. At the top, there are navigation tabs: 'Developers' (with a dropdown arrow), 'Design', 'Develop' (highlighted in orange), and 'Distribute'. To the right of these tabs are a search icon and a menu icon. Below the top navigation, there are secondary navigation links: 'Android Training' (highlighted in orange), 'API Guides', 'Reference', and 'Tools'. On the left side, there is a sidebar menu with the following items: 'Get Started' (with a dropdown arrow), 'Advanced Training' (with an up arrow), 'Making Your App Location Aware' (with a dropdown arrow), 'Performing Network Operations' (with a dropdown arrow), 'Transferring Data Without Draining the Battery' (with an up arrow and highlighted in blue), 'Optimizing Downloads for Efficient Network Access', and 'Minimizing the Effect of Regular Updates'. The main content area features the article title 'Transferring Data Without Draining the Battery' in bold. Below the title is a short paragraph: 'In this class you will learn to minimize the battery life impact of downloads and network connections, particularly in relation to the wireless radio.' This is followed by a longer paragraph: 'This class demonstrates the best practices for scheduling and executing downloads using techniques such as caching, polling, and prefetching. You will learn how the power-use profile of the wireless radio can affect your choices on when, what, and how to transfer data in order to minimize impact on battery life.' To the right of the main content, there is a box containing a 'GET STARTED >' link, a section titled 'DEPENDENCIES AND PREREQUISITES' with a bullet point 'Android 2.0 (API Level 5) or higher', and a section titled 'YOU SHOULD ALSO READ' with a bullet point 'Optimizing Battery Life'.



Google



GOAL ACHIEVED!
RELIABILITY!



Reliability

- Lint
- Strict Mode
- Test. Test. Test.



Bucket Testing

	Small	Normal	Large	Xlarge
LDPI	2.3%	0.7%	0.3%	
MDPI		26.2%	2%	7.4%
HDPI	2.4%	57.8%		
XHDPI		0.9%		



Bucket Testing

	Small	Normal	Large	Xlarge
LDPI	2.3%	0.7%	0.3%	
MDPI		26.2%	2%	7.4%
HDPI	2.4%	57.8%		
XHDPI		0.9%		



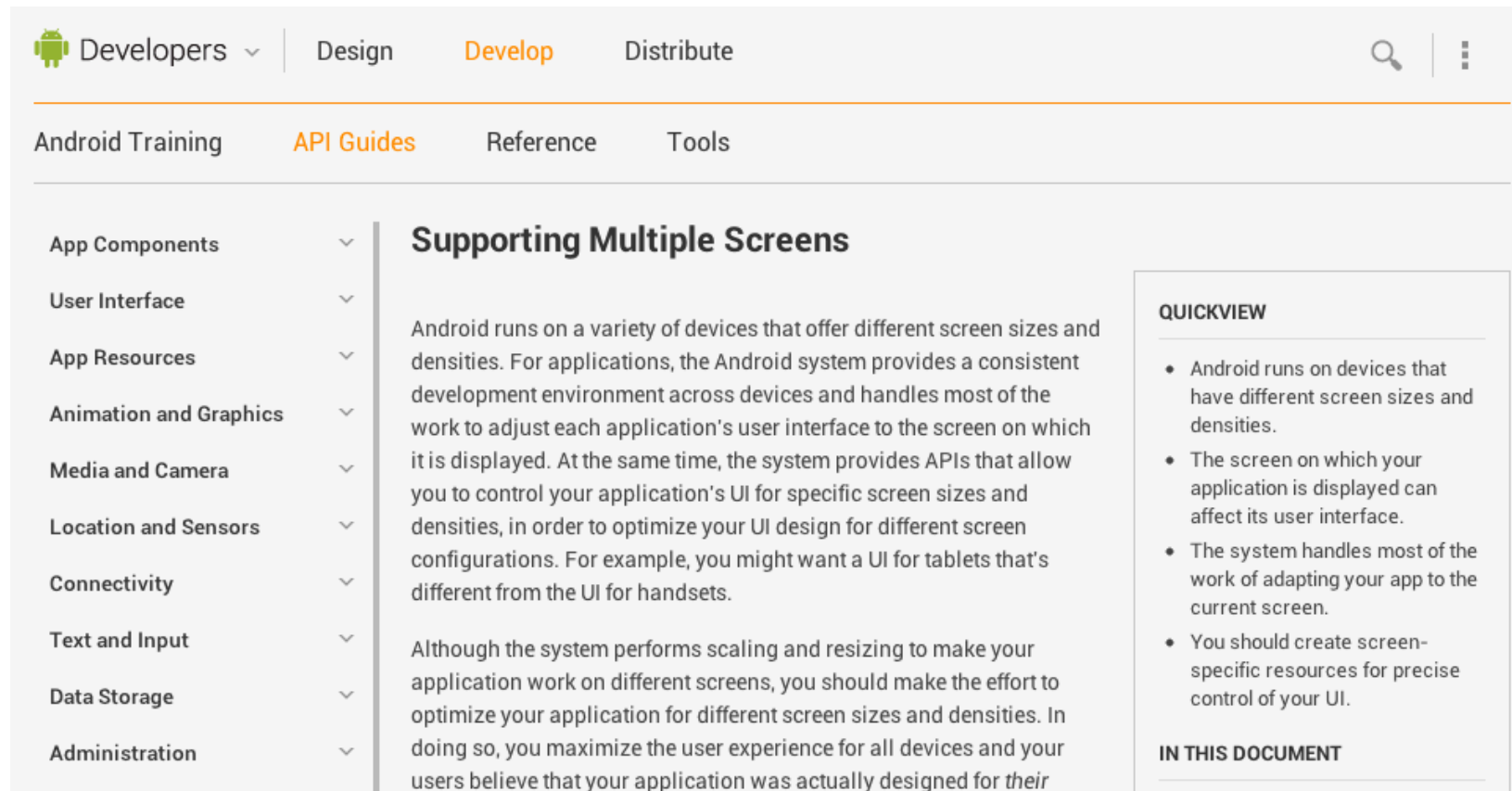
Bucket Testing

	Small	Normal	Large	Xlarge
LDPI	2.3%	0.7%	0.3%	
MDPI		26.2%	2%	7.4%
HDPI	2.4%	57.8%	Nexus 7	
XHDPI		Galaxy Nexus HTC One X Samsung S III		



Supporting Multiple Screens

http://developer.android.com/guide/practices/screens_support.html



The screenshot shows the Android Developer website interface. At the top, there is a navigation bar with the Android logo, 'Developers' dropdown, and links for 'Design', 'Develop' (highlighted), and 'Distribute'. Below this is a secondary navigation bar with 'Android Training', 'API Guides' (highlighted), 'Reference', and 'Tools'. The main content area features a left sidebar with a list of categories: App Components, User Interface, App Resources, Animation and Graphics, Media and Camera, Location and Sensors, Connectivity, Text and Input, Data Storage, and Administration. The central article is titled 'Supporting Multiple Screens'. The text explains that Android runs on various devices with different screen sizes and densities, and the system provides a consistent development environment. It also mentions that the system provides APIs to control the UI for specific screen sizes and densities. A 'QUICKVIEW' section on the right lists key points: Android runs on devices with different screen sizes and densities; the screen on which the application is displayed can affect its user interface; the system handles most of the work of adapting the app to the current screen; and you should create screen-specific resources for precise control of your UI. Below the quickview is a section titled 'IN THIS DOCUMENT'.

Developers ▾ | Design | **Develop** | Distribute

Android Training | **API Guides** | Reference | Tools

App Components ▾ | **Supporting Multiple Screens**

User Interface ▾

App Resources ▾

Animation and Graphics ▾

Media and Camera ▾

Location and Sensors ▾

Connectivity ▾

Text and Input ▾

Data Storage ▾

Administration ▾

Android runs on a variety of devices that offer different screen sizes and densities. For applications, the Android system provides a consistent development environment across devices and handles most of the work to adjust each application's user interface to the screen on which it is displayed. At the same time, the system provides APIs that allow you to control your application's UI for specific screen sizes and densities, in order to optimize your UI design for different screen configurations. For example, you might want a UI for tablets that's different from the UI for handsets.

Although the system performs scaling and resizing to make your application work on different screens, you should make the effort to optimize your application for different screen sizes and densities. In doing so, you maximize the user experience for all devices and your users believe that your application was actually designed for *their*

QUICKVIEW

- Android runs on devices that have different screen sizes and densities.
- The screen on which your application is displayed can affect its user interface.
- The system handles most of the work of adapting your app to the current screen.
- You should create screen-specific resources for precise control of your UI.

IN THIS DOCUMENT



Stability

- Monkey Runner
- Alpha testers
- Emulator





Android: Smartphones++









“I don't use the Internet on my phone...”



“I don’t use the Internet on my phone.
Of course I use <insert your app here>.”

Your users



Thank You!

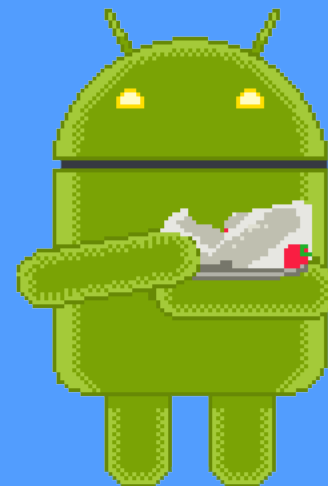
Questions? Join the Android Developer Relations team every Wednesday, 2pm Pacific (UTC-7) for our live Office Hours Q&A sessions at developers.google.com/live

Graphics & animations by Pandamus (@pandamus)

+Reto Meier | @retomeier

<http://developer.android.com/+>

#IO12





Google
Developers