

안드로이드 오픈소스 어플리케이션 블럭



EVANGELIST
Architecture Design



NHN NEXT
손영수



KTH
진성주



말랑 스튜디오
(김영호, 강진석)



딜라이트룸
신재명

Google이
꿈꾸는..

Android의 미래

Google은 왜 modu의 특허를 50억에 샀을까?

noun. the first ever modular phone *verb.* 1
to choose freely 2 to alter appearance 3 to bring
about change *adjective.* flexible, stylish, exciting,
personal, expressive



modu의 Jacket 시스템

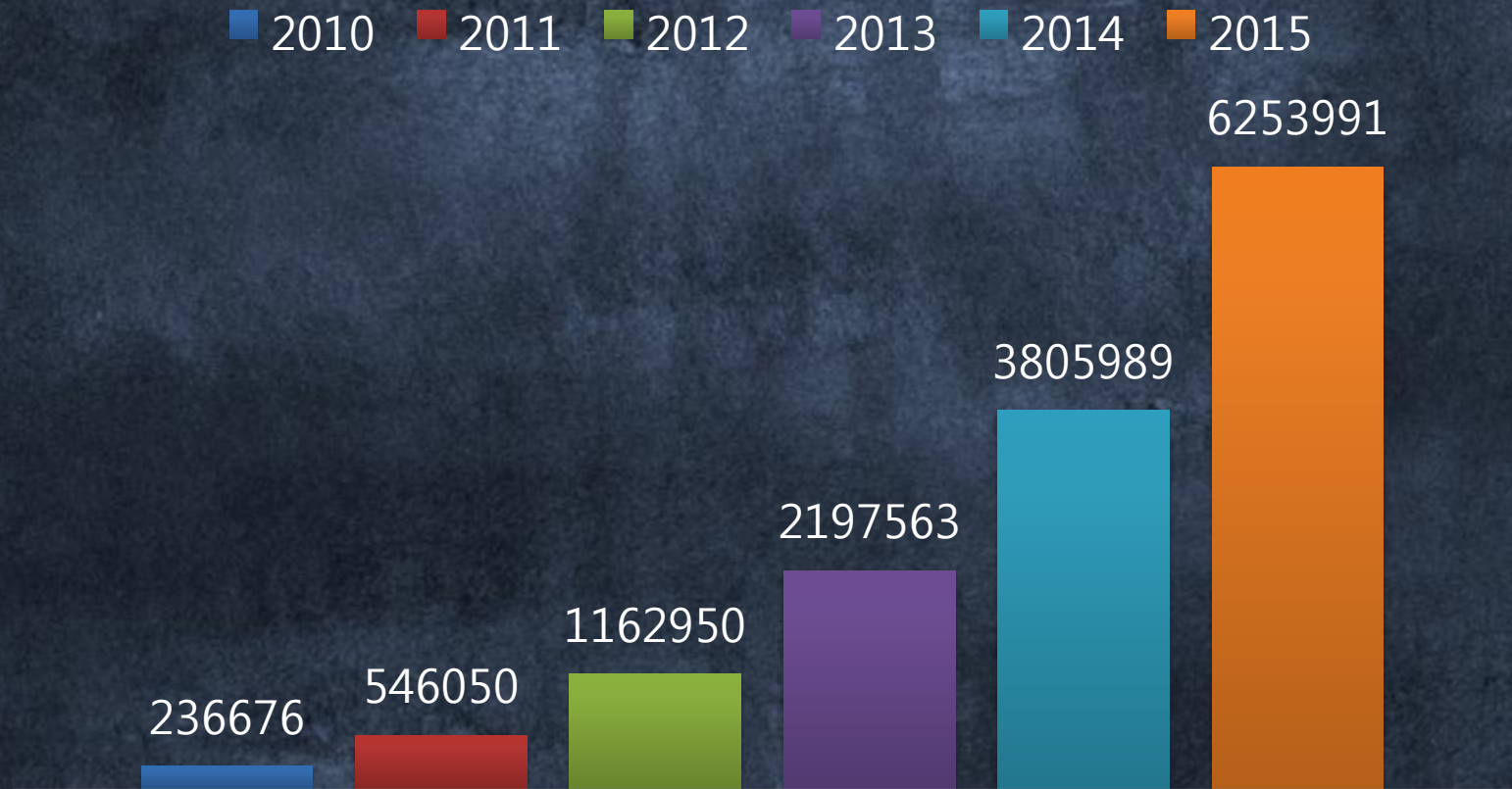


ANDROID



이게
왜 중요하지?..

갈수록 증가하는 모바일 Traffic

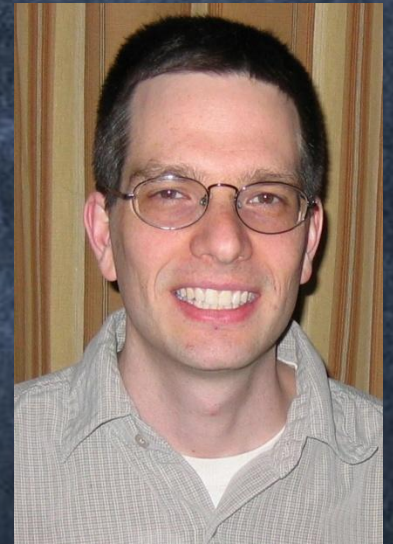
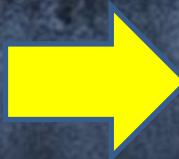


Cisco (2012), Cisco Visual Networking Index

단위 TB/월

사실 Google의
서비스는 Cloud에..

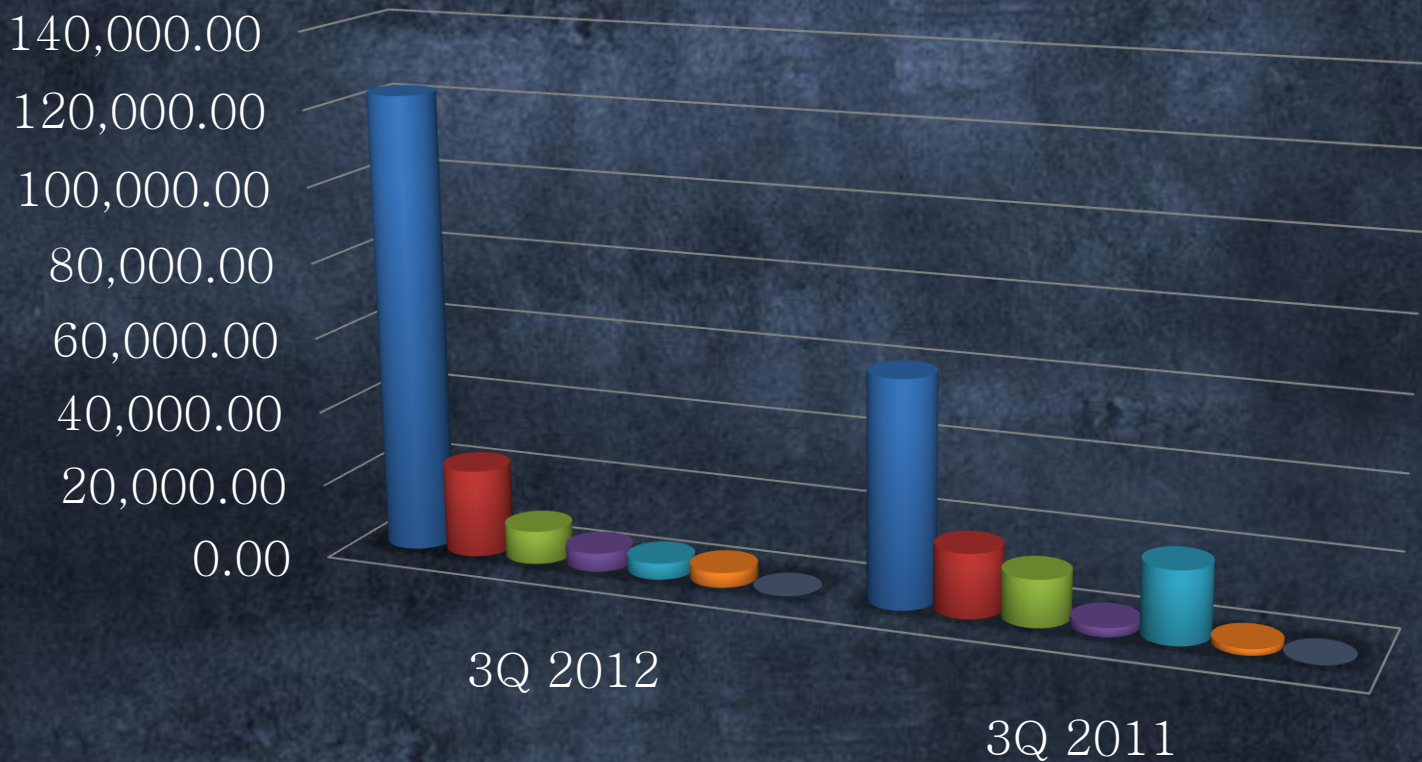






Android는 Google의 서비스를
(개인, 가정이) 소비하게 만드는 Hub

시장 점유율 (52% -> 72%)



■ Android ■ iOS ■ RIM ■ Bada ■ Symbian ■ Microsoft ■ Others

Gartner Smartphone M/S- <http://on.mash.to/TW5EAA>

또 한...

Android는
더 이상
Gadget을 위한
OS가 아닙니다.

Embedded
System을
포괄할 OS로 봐야 함.



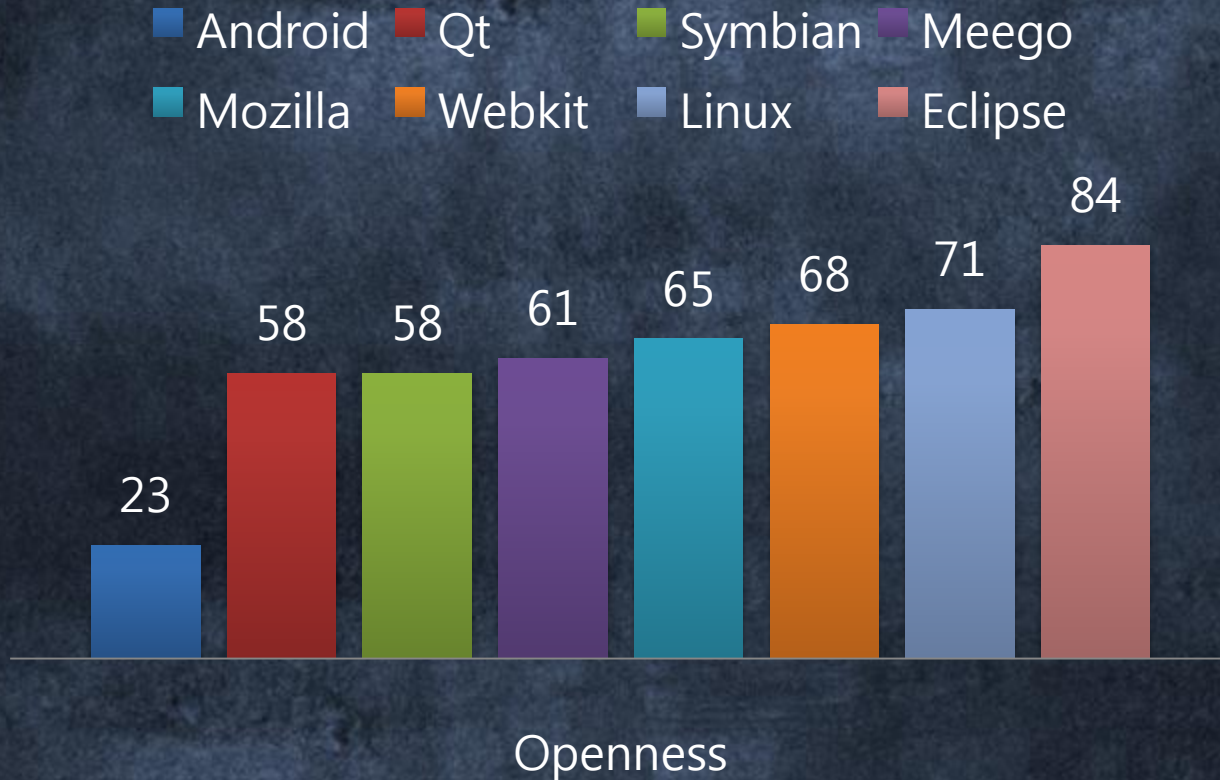
하지만..





제조사 입장에서
Android는
독이 든 성배..

안드로이드의 개방성



Vision Mobile Report July, 2011-
<http://bit.ly/oTmEmF>

오픈 소스 이긴 한데.. 구글러가 아니면 할수 있는게 없다.

Best practices

Ease of source-code access via the Apache License

Ease of access to mailing lists, very good developer tools and forums

Simple code-contributions process for developers to follow

Clever targeting of developers via the Android Challenge, Summer of Code, etc.

Practices to avoid

Unilateral Android project decision-making processes, as Google determines the roadmap, feature-set and releases of Android

Closed code committer process, i.e., committers are exclusively Google personnel

Closed contributions process model

Opaque decision-making and control process around the Android Compliance Program

No project metrics around contributions, commits, contributors, top participants and bugs

No public information provided regarding meeting minutes or decisions.

No intention to move towards a more open governance model

정리하면..

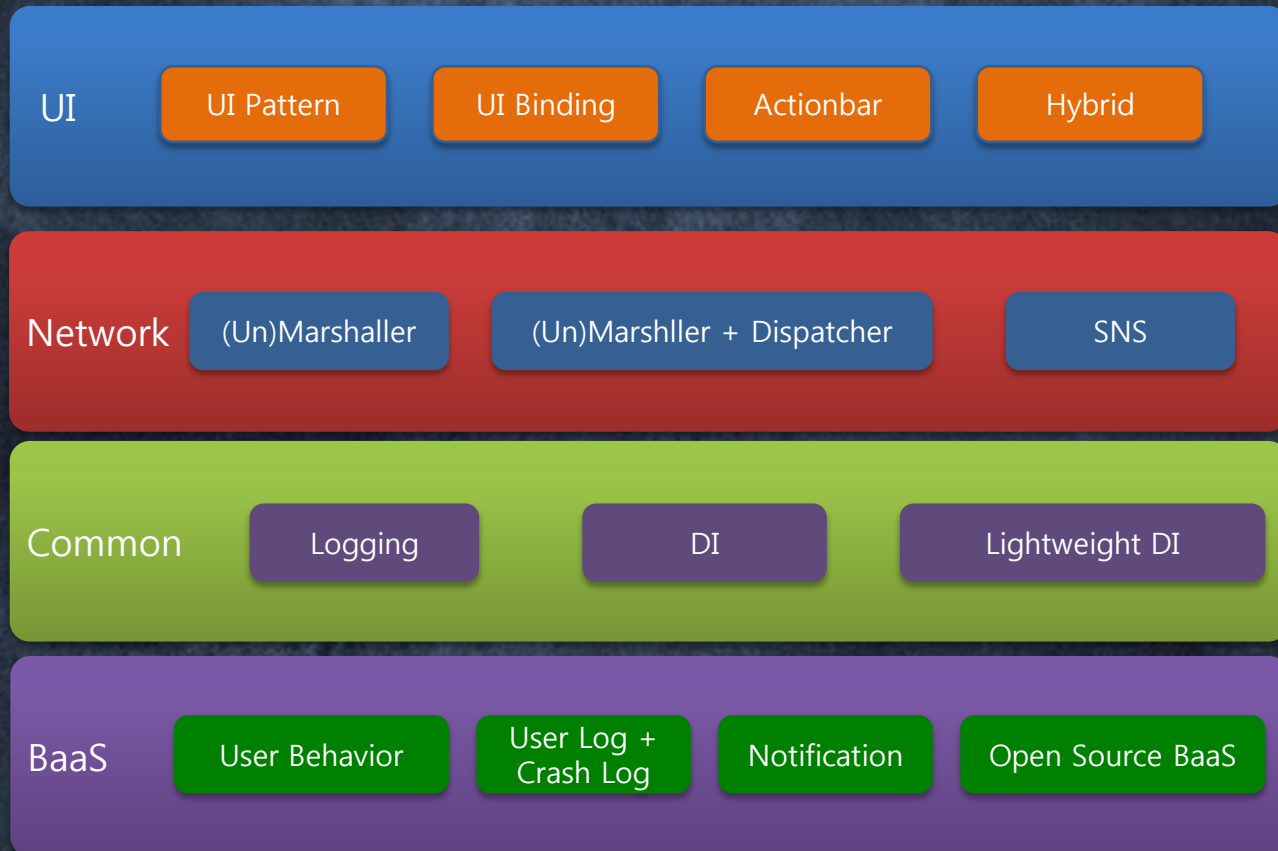
버그만 잡아줘..





어플리케이션 블럭이란..

Android Application Block

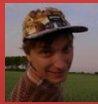


Android의 단점을 극복할 대체제들을 잘써야 한다.

UI



android-binding



android-actionbar



actionbar shellok



android ui
patterns

Network



gson (json)



Simple
(REST)



Facebook (fhalo)

Common



Android billing
(In app purchase)



roboguice



microlog
4android



sensing

BaaS



User Behavior
Analytics



LogDog



Urban Airship

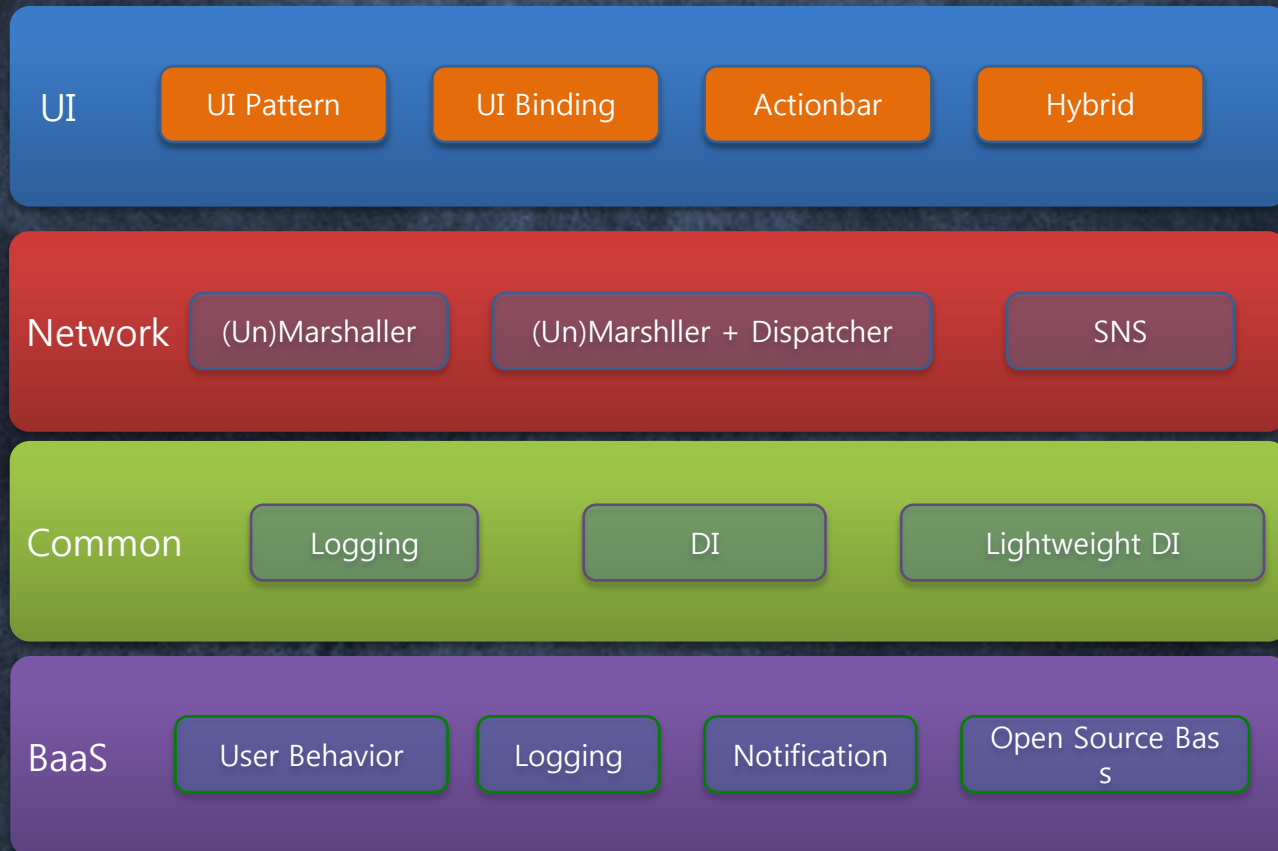


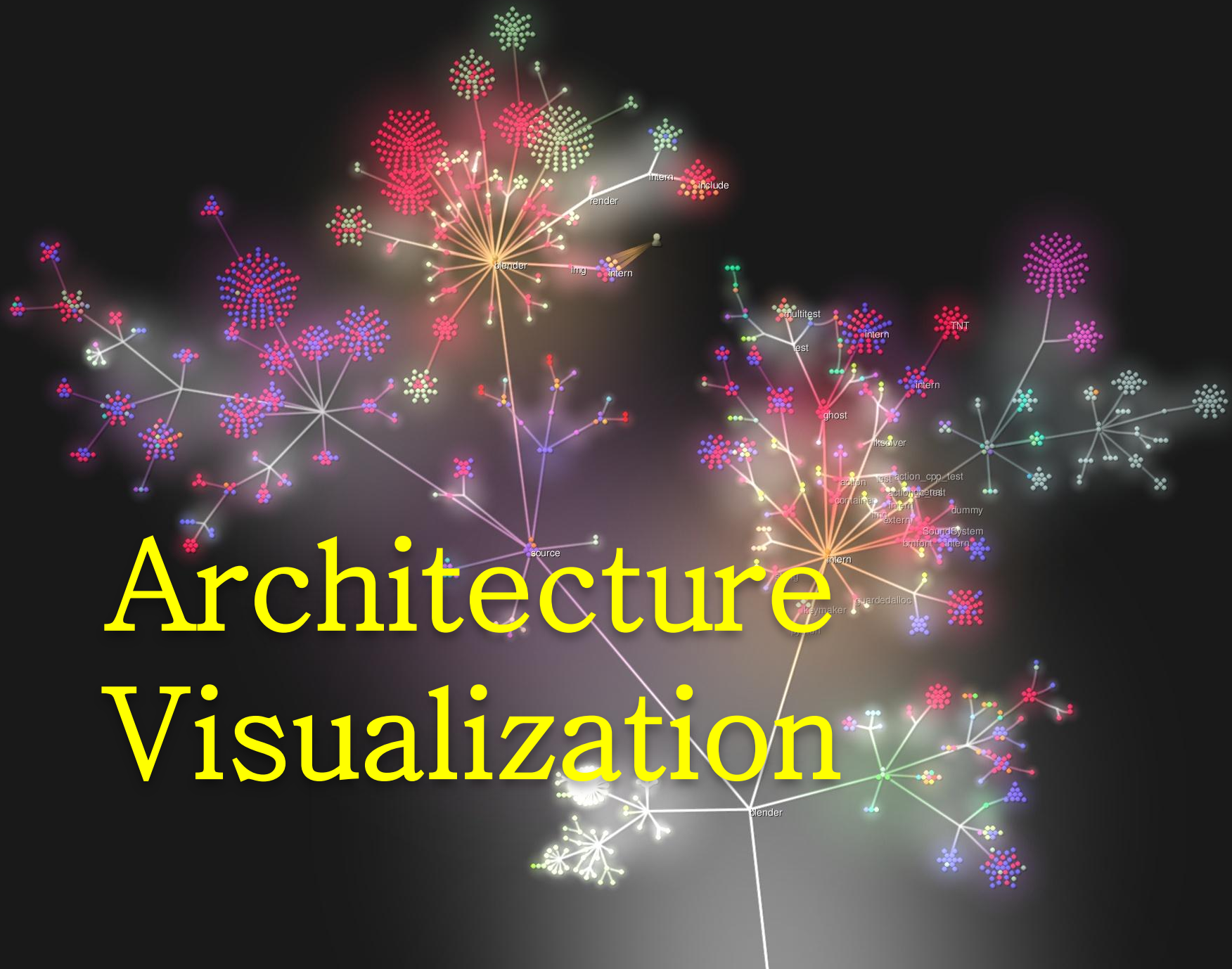
USERGRID
(BaaS Open Source)

Welcome to my Journey!



Part I. UI Layer





높이 (30000 feet) 봐야 할까?



푸기노이치나요??

자세히
(0 feet) 봐야 할까?

FOLDERS

work

0518

ruby

monitor.py

transform.py

transform.py

```
1 #!/usr/bin/python
2 from __future__ import nested_scopes
3
4 import os, time, re, csv
5
6 def watch_directories(paths, func, delay=1.0):
7
8
9     all_files = {}
10     def f(unused, dirname, files):
11         for filename in files:
12             path = os.path.join(dirname, filename)
13
14             try:
15                 t = os.stat(path)
16             except os.error:
17                 # If a file has been deleted between os.path.walk()
18                 # scanning the directory and now, we'll get an
19                 # os.error here. Just ignore it -- we'll report
20                 # the deletion on the next pass through the main loop.
21                 continue
22
23             mtime = remaining_files.get(path)
24             if mtime is not None:
25                 # Record this file as having been seen
26                 del remaining_files[path]
27                 # File's mtime has been changed since we last looked at it.
28                 if t.st_mtime > mtime:
29                     added_list.append(path)
30             else:
31                 # No significant change, so it's either
32                 # a brand new file, or
33                 added_list.append(filename)
34
35             # Record current mtime of f
36             all_files[filename] = t.st_mtime
37
38     # Main loop
39
40     rescan = False
41     while True:
42         added_list = []
43         remaining_files = all_files.copy()
44         all_files = {}
45         for path in paths:
46             os.path.walk(path, f, None)
47         if rescan:
48             rescan = False
49         elif added_list:
50             rescan = func(added_list)
51
52         time.sleep(delay)
53
54
55 if __name__ == '__main__':
56     def f(added_files):
57         filename = added_files[0]
58         filetext = re.split(r'[_]+', filename)
59         sitenum = filetext[0]
60         numrecs = filetext[1]
61         datafile = filetext[2]
62         dirname = '/data/testing/upload/%s' % (sitenum)
63
64         try:
65             os.makedirs(dirname)
66         except OSError:
67             if os.path.exists(dirname):
68                 pass
69             else:
70                 raise
```

monitor.py

```
1 #!/usr/bin/python
2
3 import os
4 import time
5 import re
6 import errno
7 import csv
8
9 path_to_watch = "/data/testing/uploads"
10 before = dict([(f, None) for f in os.listdir(path_to_watch)])
11 while 1:
12     time.sleep(5)
13     after = dict([(f, None) for f in os.listdir(path_to_watch)])
14     added = [f for f in after if not f in before]
15     removed = [f for f in before if not f in after]
16     if added:
17         print "Added: ", ", ".join(added)
18         filetext = f
19         del filetext
20         upload = f
21         filetext = re.split(r'[_]+', f)
22         sitenum = filetext[0]
23         numrecs = filetext[1]
24         filename = filetext[2]
25         print sitenum
26         print numrecs
27         print filename
28         dirname = '/data/testing/upload/%s' % (sitenum)
29         try:
30             os.makedirs(dirname)
31         except OSError:
32             if os.path.exists(dirname):
33                 pass
34             else:
35                 raise
36
37     destfile = '/data/testing/uploads/%s.%s.xml' % (sitenum, filename)
38     try:
39         open(destfile, 'a')
40     except OSError:
41         if os.path.exists(destfile):
42             pass
43         else:
44             raise
45
46     csvFile = '/data/testing/uploads/%s' % (upload)
47     xmlFile = '/data/testing/uploads/%s/%s.xml' % (sitenum, filename)
48     csvData = csv.reader(open(csvFile))
49     xmlData = open(xmlFile, 'w')
50     xmlData.write('<?xml version="1.0"?>' + "\n")
51     xmlData.write('<csv_data>' + "\n")
52
53     rowNum = 0
54     for row in csvData:
55         if rowNum == 0:
56             tags = row
57             # replace spaces w/ underscores in tag names
58             for i in range(len(tags)):
59                 tags[i] = tags[i].replace(' ', '_')
60
61             else:
62                 xmlData.write('<row>' + "\n")
63                 for i in range(len(tags)):
64                     xmlData.write('<' + tags[i] + '>' + \
65                                   row[i] + '<' + tags[i] + '>' + "\n")
66                 xmlData.write('</row>' + "\n")
67
68     rowNum = rowNum + 1
```

제한된 코드로 프로젝트가
잘되고 있는지 판단은?

3만 피트 vs 0 피트의 뷰.



3만 피트

- 다이어그램의 Line의 의미는?
- 의존성?
- 데이터 흐름?
- 버스와 같은 공유자원?



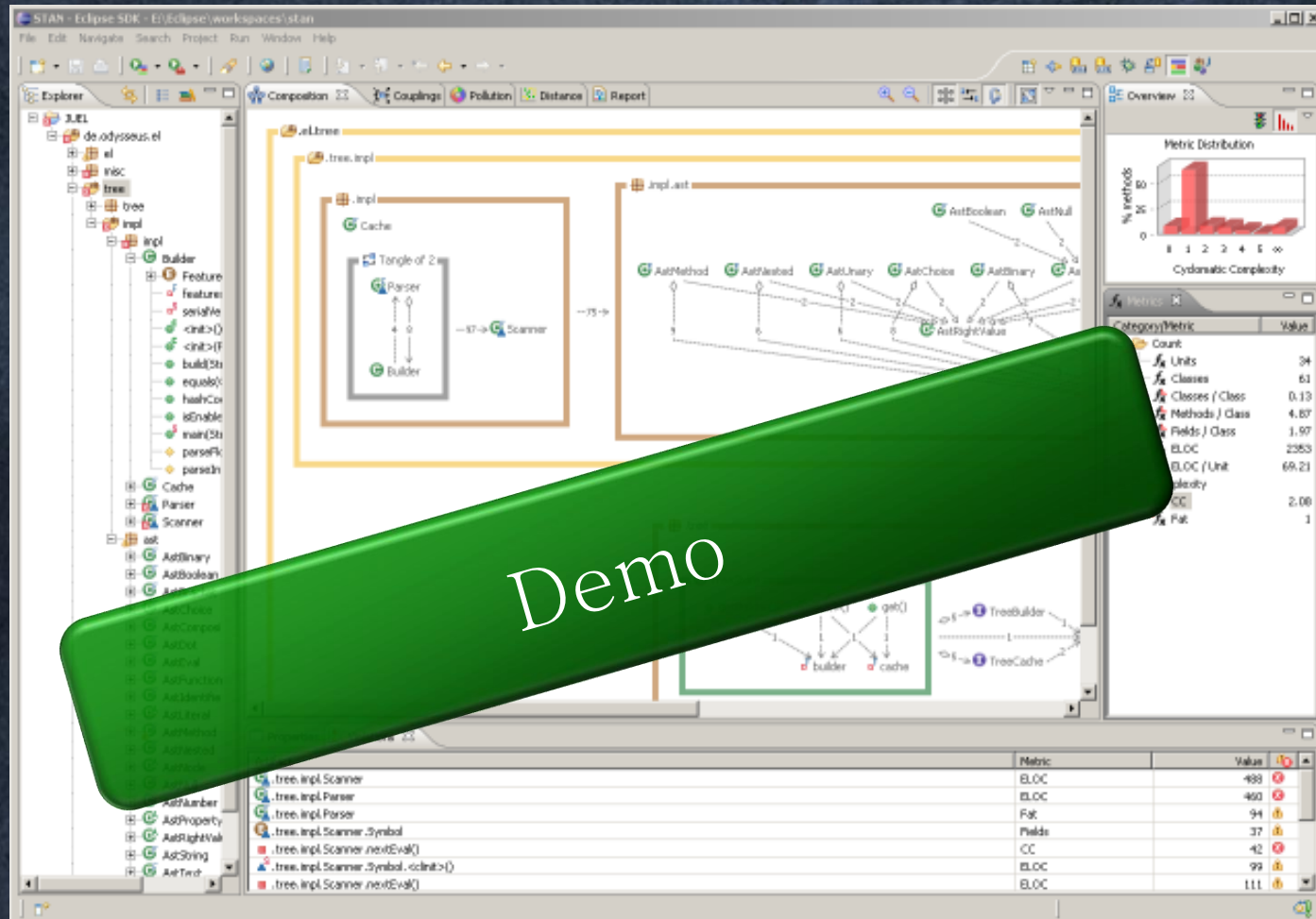
0 피트

- 너무 상세한 정보임.
- 전체적인 구조를 보지 못함.



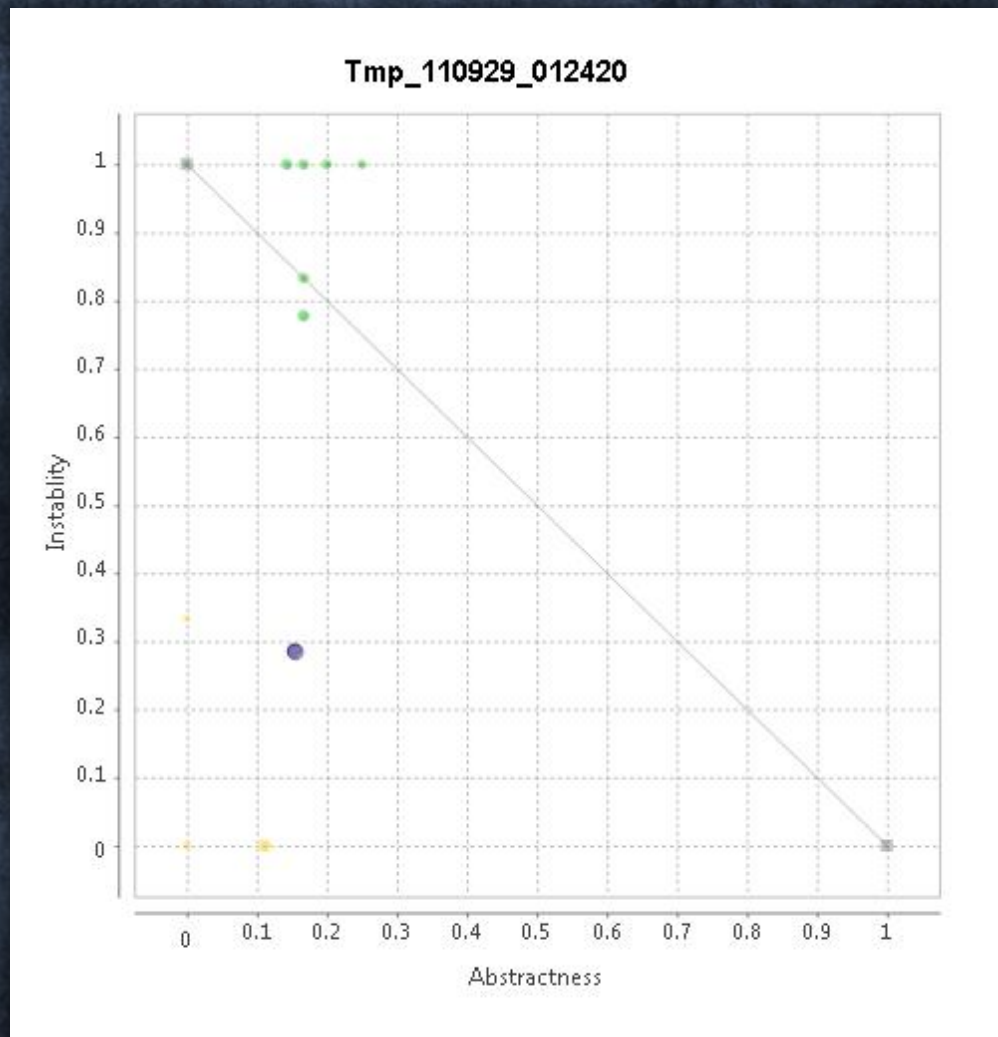
해결책은..
적절한 1000 피트의 뷰

STAN (Structure Analysis for Java)



STAN - <http://stan4j.com/eclipse/eclipse-integration.html>

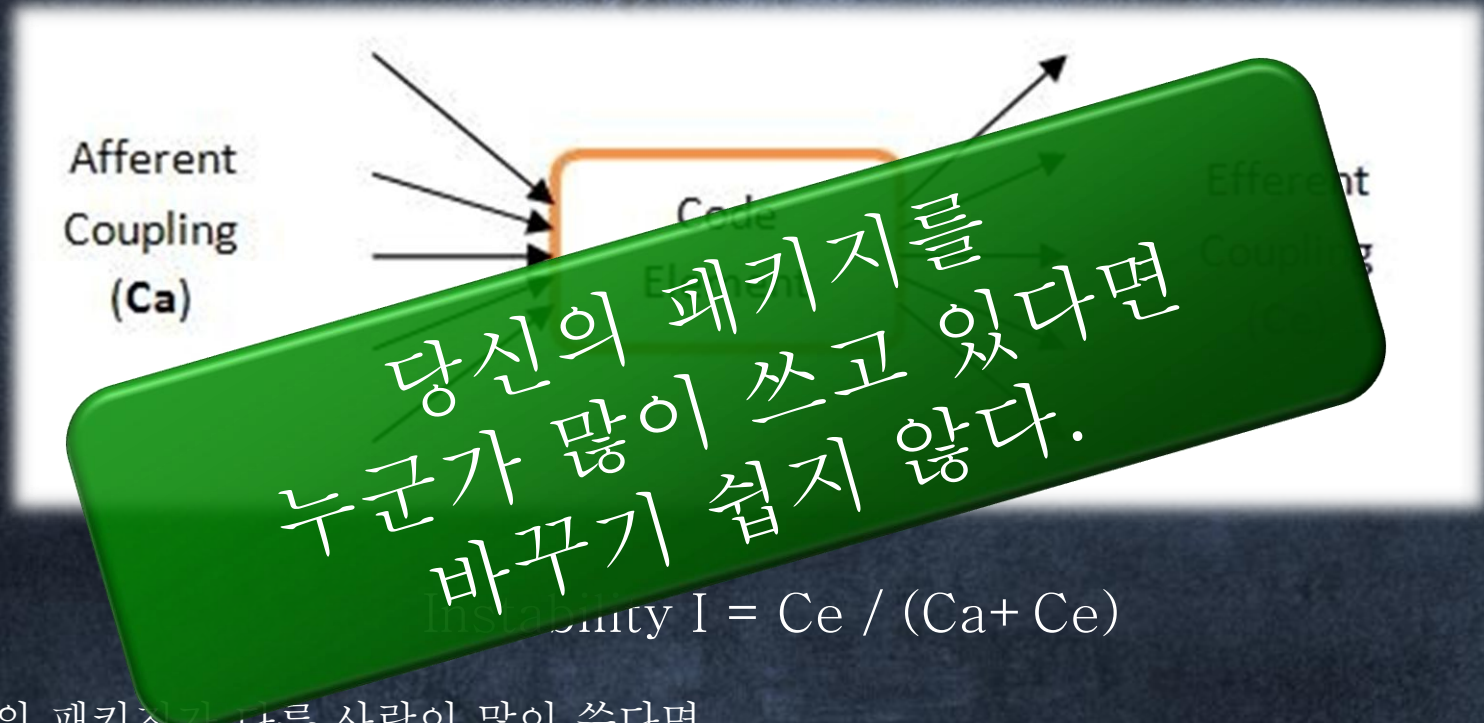
Robert C. Martin의 그래프



Instability

- 패키지의 움직임 수 있는 여력을 판단하는 지표
- 다른 패키지에 영향을 미치지 않고,
해당 패키지를 쉽게 변경 할 수 있는가?
- Instability $I = Ce / (Ca + Ce)$
- Ce = Efferent Coupling (Ingoing Dependencies)
- Ca = Afferent Coupling (Outgoing Dependencies)

Instability



당신의 패키지가 다른 사람이 많이 쓴다면,
즉 Outgoing, C_a 가 많다면, 여러분의 패키지는 변경하기 어렵다.

반대로 Outgoing하는 C_a 가 적고, Ingoing(다른 패키지만 사용만 하는) C_e ,
여러분의 패키지는 쉽게 변경해도 된다.

즉 0.0에서 0.3이면 안정적인 버전, 0.7에서 1.0이면 불안정적인 상태다

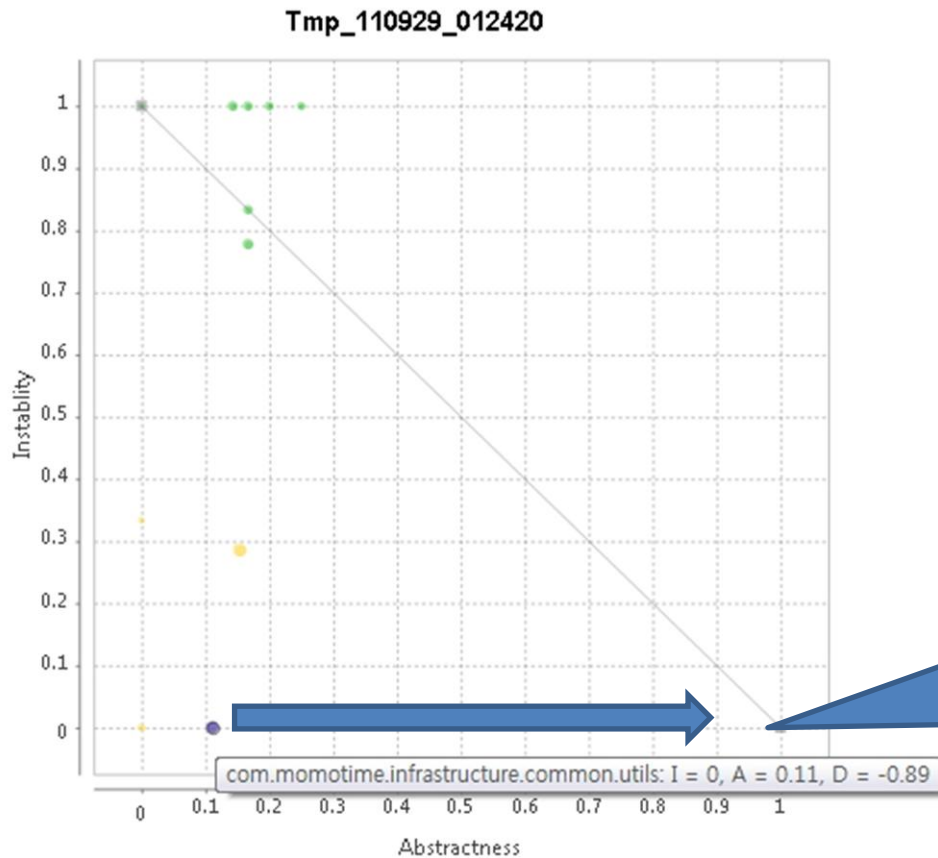
Abstractness

Interface(Abstract) 와 Concrete Class를 비교

$A = (\text{\#abstract classes} / \text{total \# of classes})$

- Abstract class = interface, abstract다 포함
- Total # class = abstract class + concrete class
- 0 이면 concrete class만 있다.
- 1 이면 abstract class만 있다.

다시 보는 그래프



다른데서 많이 쓰는
녀석이니 조금 더
abstract를 높여야
돼!

그 외 용어

- Tangled Complexity
 - 순환 참조가 있어 Boundary를 깰 때
- Cyclomatic Complexity
 - 분기 문이 많아 hotspot이 될 가망성이 높은 곳

경고!!!

환자의 외부 증상만
고치는 의사가 되지 말자!!

이러한 정보는 좋은 가이드일뿐!!

숫자에 의존하다가
오히려 문제가 보이지 않게 된다.

1장. UI Patterns

다양한 안드로이드 UI 오픈소스들을 소개

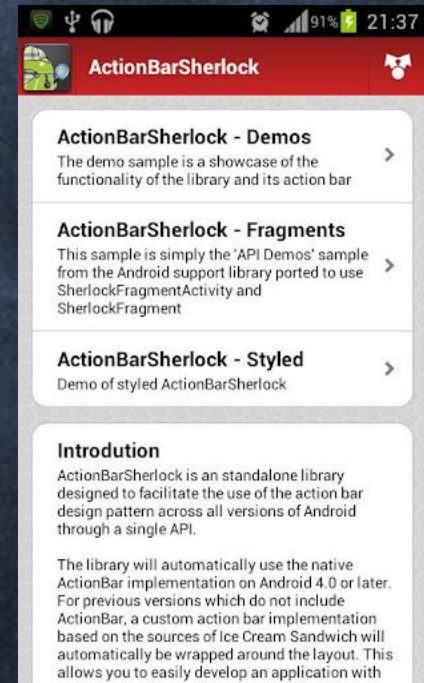
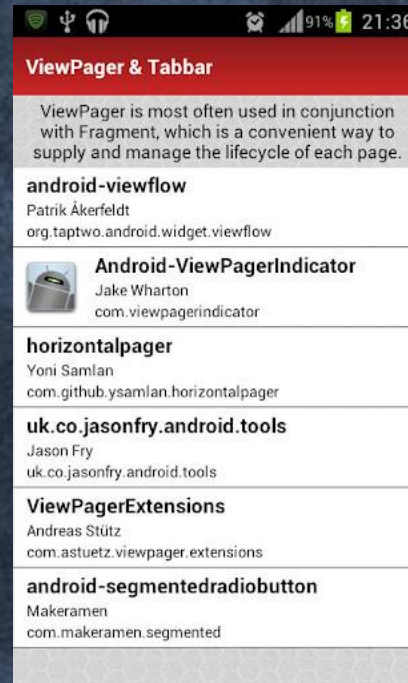
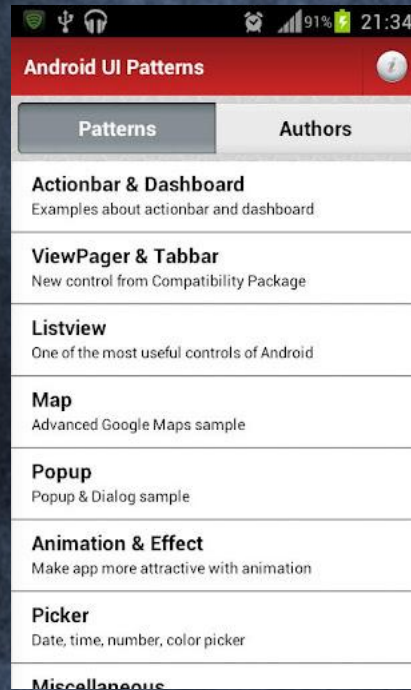
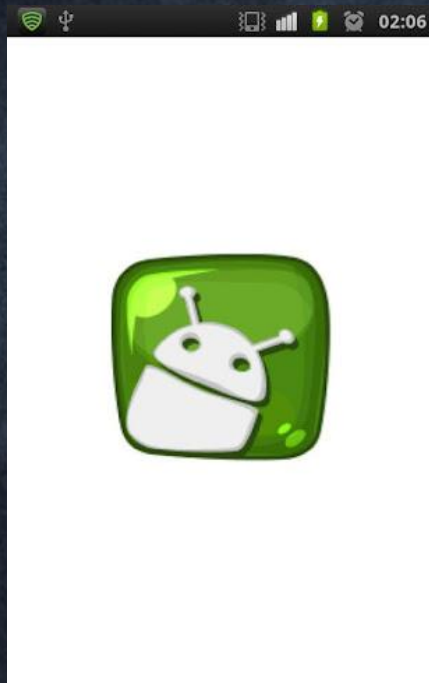
<http://bit.ly/QyDjMV>

안드로이드 ScreenCast

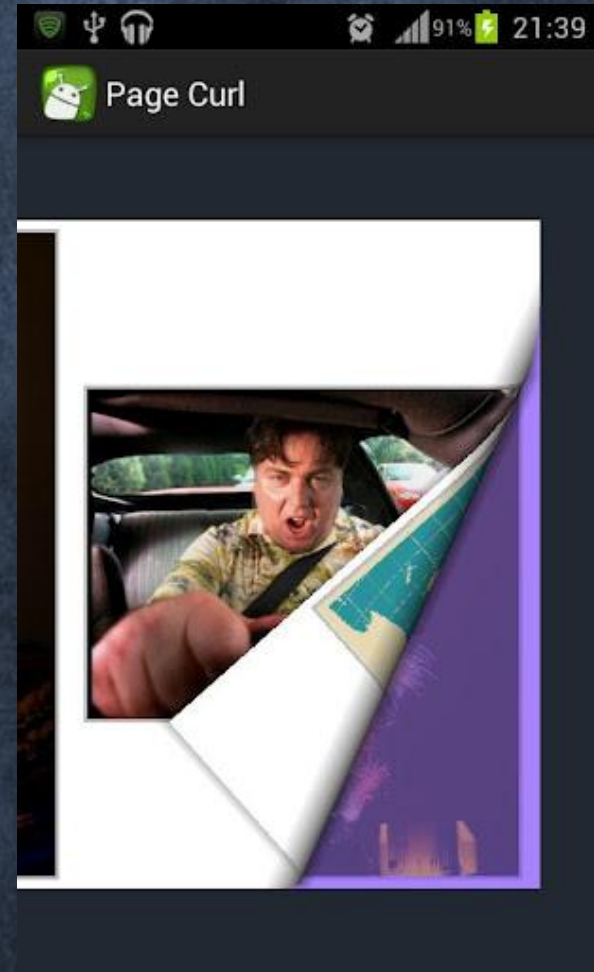
<http://bit.ly/cOzoD2>

단편화
+
고급적인 효과..

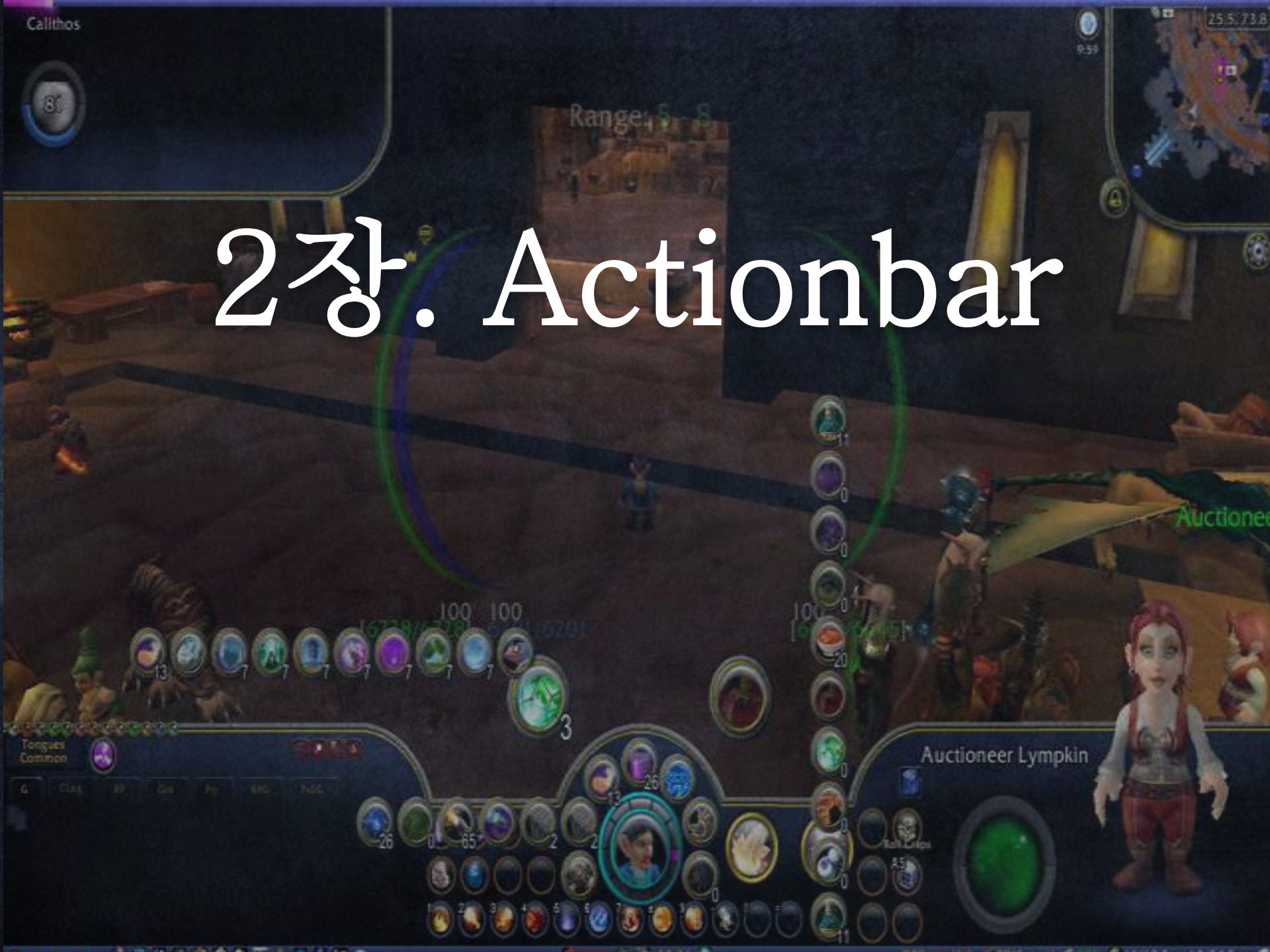
Android UI Patterns App



Android UI Patterns App



2장. Actionbar





Other



스크린샷1

스크린샷2

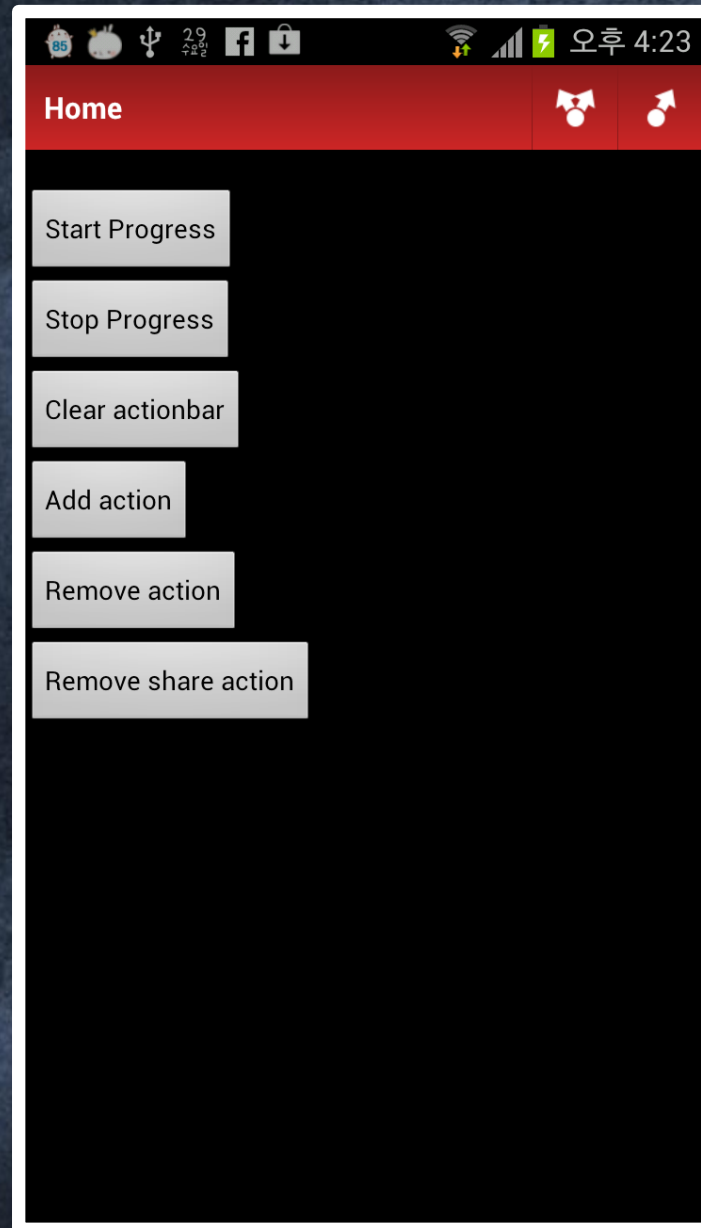
XML

기능

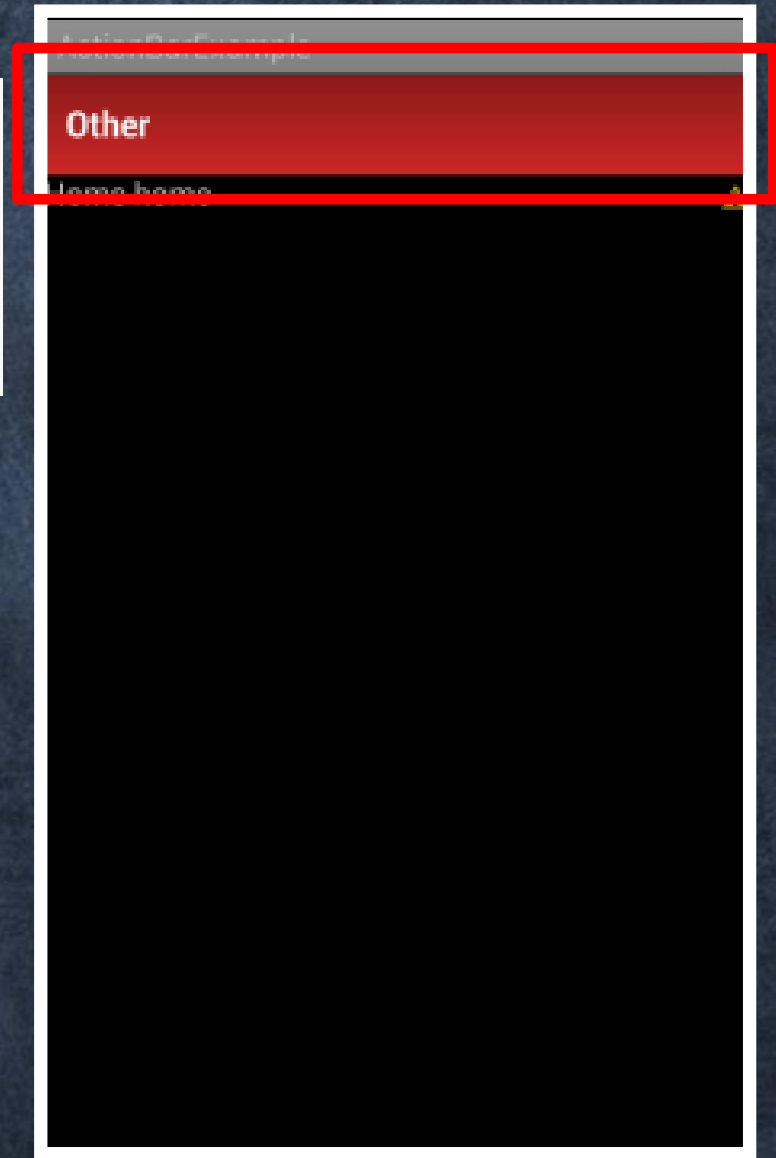
Action

Action예제

색변경




```
<com.markupartist.android.widget.ActionBar  
    android:id="@+id/actionbar"  
    app:title="@string/some_title"  
    style="@style/ActionBar"  
/>
```



```
final ActionBar actionBar = (ActionBar) findViewById(R.id.actionbar);  
//actionBar.setHomeAction(new IntentAction(this, createIntent(this), R.drawable.actionbar));  
actionBar.setTitle("Home");
```

```
final Action shareAction = new IntentAction(this, createShareIntent(), R.drawable.actionbar);  
actionBar.addAction(shareAction);  
final Action otherAction = new IntentAction(this, new Intent(this, OtherActivity.class), R.drawable.actionbar);  
actionBar.addAction(otherAction);
```




```
final Action shareAction = new IntentAction(this, createShareIntent(), R.drawable.ic_ti
```

```
public interface Action {  
    public int getDrawable();  
    public void performAction(View view);  
}
```

```
private class ToastAction implements Action {

    @Override
    public int getDrawable() {
        return R.drawable.ic_title_export_default;
    }

    @Override
    public void performAction(View view) {
        Toast.makeText(OtherActivity.this,
            "Example action", Toast.LENGTH_SHORT).show();
    }

}
```


colors.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="actionbar_separator">#3A5FCD</color>
    <color name="actionbar_background_start">#3A5FCD</color>
    <color name="actionbar_background_end">#27408B</color>
</resources>
```



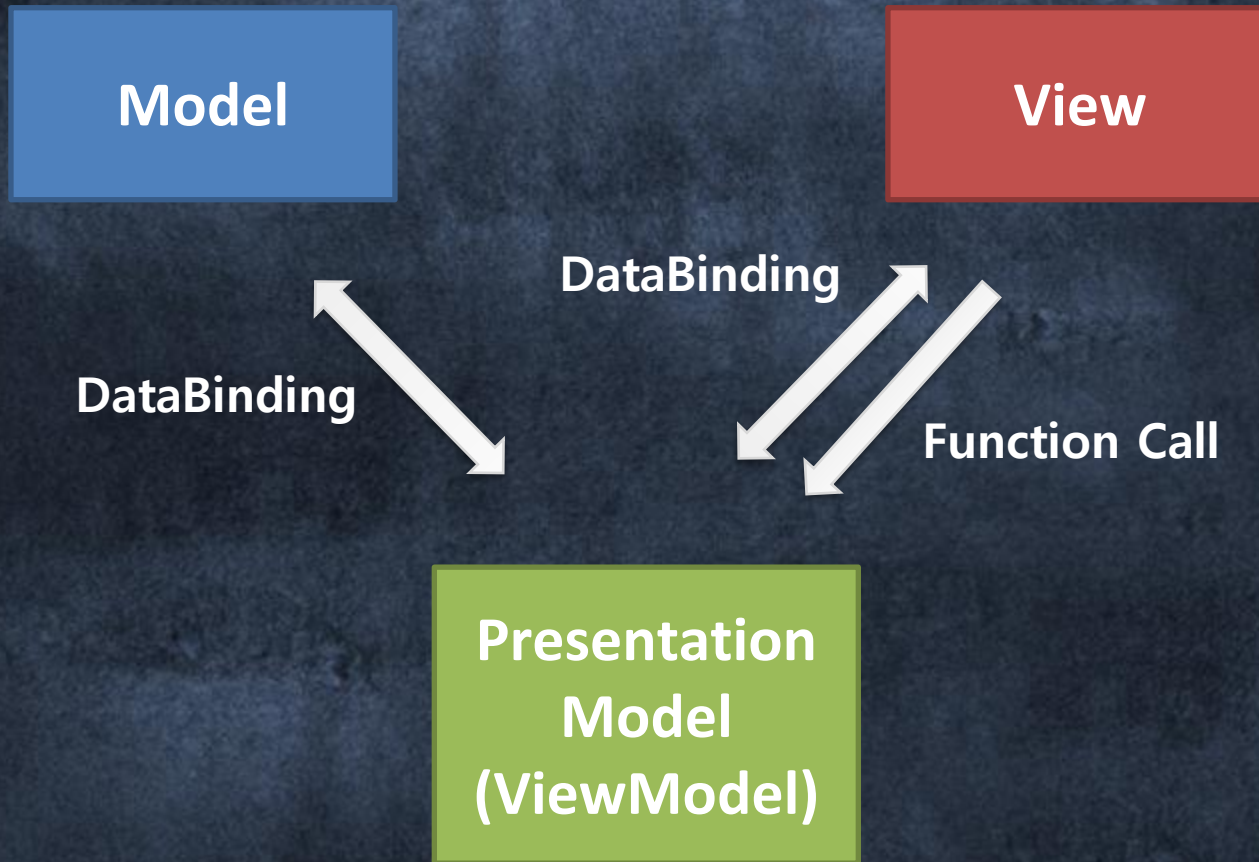
3장. Binding Framework



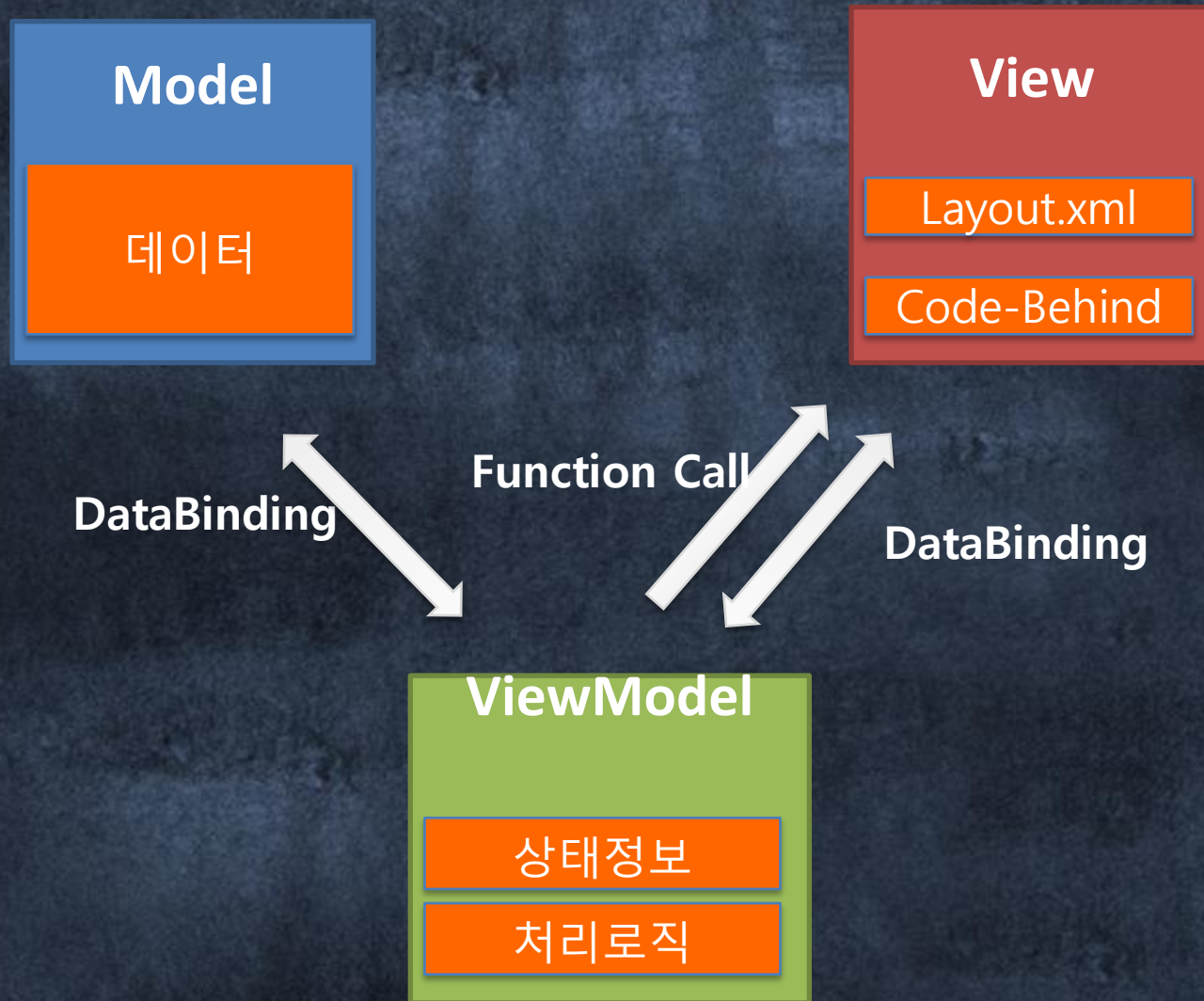
<http://code.google.com/p/android-binding/>

<http://bit.ly/cxxwCF>

MVVM

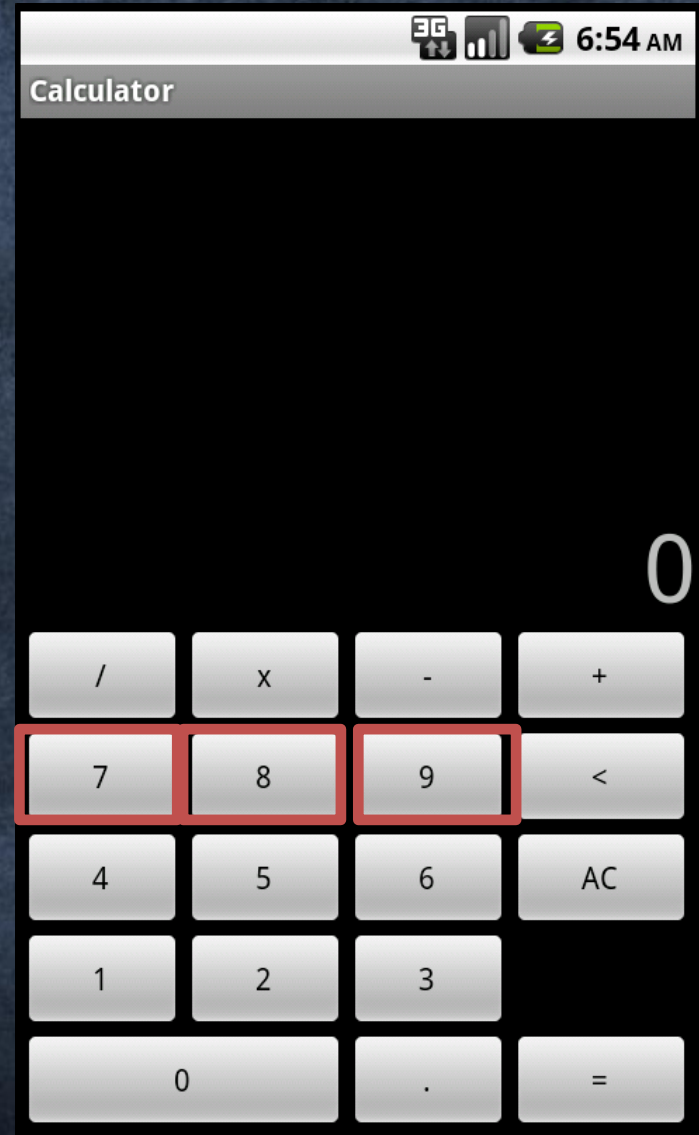


MVVM in Android Binding



Event에 대한 Binding

다음과 같이 버튼에 대한 이벤트
Method OnClick()



Main.xml

```
<TableRow>
    <Button android:text="/" binding:onClick="Divide" />
    <Button android:text="x" binding:onClick="Multiply" />
    <Button android:text="-" binding:onClick="Minus" />
    <Button android:text="+" binding:onClick="Plus" />
</TableRow>
<TableRow>
    <Button android:text="7" binding:onClick="Number7" />
    <Button android:text="8" binding:onClick="Number8" />
    <Button android:text="9" binding:onClick="Number9" />
    <Button android:text="&lt;" binding:onClick="Back" />
</TableRow>
<TableRow>
    <Button android:text="4" binding:onClick="Number4" />
    <Button android:text="5" binding:onClick="Number5" />
    <Button android:text="6" binding:onClick="Number6" />
    <Button android:text="AC" binding:onClick="AllClear" />
</TableRow>
```


CalculateViewModel.java

```
private class NumberCommand extends Command{
    private int mNumber;
    public NumberCommand(int number){
        mNumber = number;
    }
    public void Invoke(View arg0, Object... arg1) {
        addNumber(mNumber);
    }
}

public NumberCommand Number9 = new NumberCommand(9);
public NumberCommand Number8 = new NumberCommand(8);
public NumberCommand Number7 = new NumberCommand(7);
public NumberCommand Number6 = new NumberCommand(6);
public NumberCommand Number5 = new NumberCommand(5);
public NumberCommand Number4 = new NumberCommand(4);
public NumberCommand Number3 = new NumberCommand(3);
public NumberCommand Number2 = new NumberCommand(2);
public NumberCommand Number1 = new NumberCommand(1);
public NumberCommand Number0 = new NumberCommand(0);
```

Calculator

0

/

x

-

+

7

8

9

<

4

5

6

AC

1

2

3

0

to advanced mode

=

Option.xml

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android"
      xmlns:binding="http://www.gueei.com/android-binding/"
>
    <item
        android:id="@+id/options1"
        binding:title="IF(AdvancedMode, 'switch to simple mode', 'to advanced mode')"
        binding:onClick="ToggleMode"/>
</menu>
```

Main Activity

```
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);

    if (getIntent().getBooleanExtra(MODE, false)){
        AdvancedMode.set(true);
    }

    ViewModel = new CalculatorViewModel();

    // unwrap the view model if exists
    Bundle parcel_model = getIntent().getBundleExtra(MODE);
    if (parcel_model!=null){
        ViewModelParceler.restoreViewModel(parcel_model, ViewModel);
    }

    // Bind the Root View with the Calculator view model
    if (AdvancedMode.get())
        setAndBindRootView(R.layout.advanced, ViewModel);
    else
        setAndBindRootView(R.layout.main, ViewModel);

    // Bind the options menu with activity
    this.setAndBindOptionsMenu(R.menu.options, this);
}

public final BooleanObservable AdvancedMode = new BooleanObservable(false);
```


Main Activity

```
public final Command ToggleMode = new Command(){
    @Override
    public void Invoke(View arg0, Object... arg1) {
        // Recreate the Activity
        Intent intent = new Intent(Calculator.this, Calculator.class);
        intent.putExtra(MODE, !AdvancedMode.get());

        // Also package the view model state to the future self
        intent.putExtra(MODEL, ViewModelParceler.parcelViewModel(ViewModel));

        Calculator.this.finish();
        Calculator.this.startActivity(intent);
    }
};
```



7:11 AM

Calculator

0

sin

cos

tan

log

/

x

-

+

7

8

9

<

4

5

6

AC

1

2

3

0

.

=

4장.

Hybrid Framework

Native

???

Web





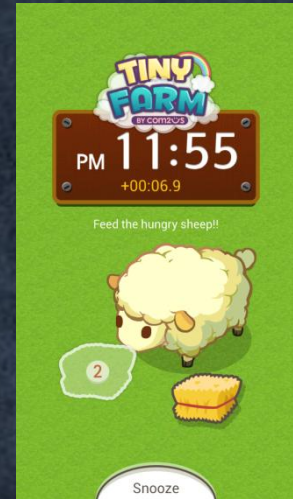
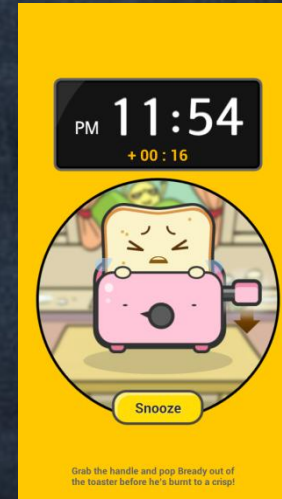
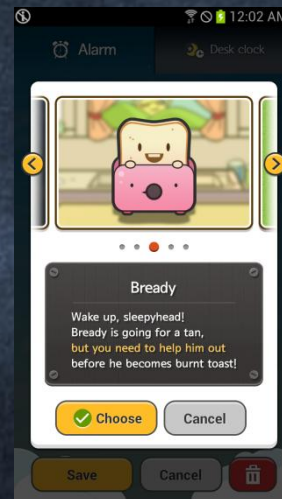
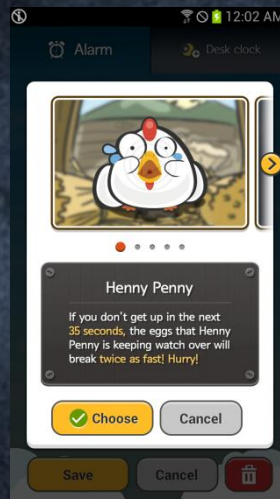
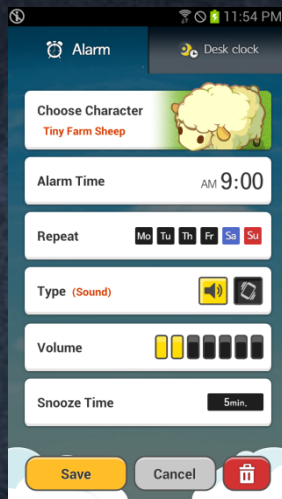
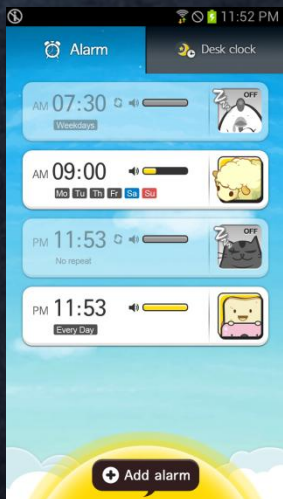
I'm ready to wake you up!



choose me boss!

130만 명의 아침을 깨운 '알람몬'

- 다양한 알람몬들과 Game으로 시작하는 즐거운 아침!
- 모든 해상도 대응
- 한국어, 일본어, 영어, 중국어 지원



해상도 파편화 문제!!

Resolution

320x480, 480x800, 540x960, 768x1024

720x1280,

800x1280

1600x2560

Cocos2D 적용 전

Xml 레이아웃 3종류 만들기

- layout-hdpi : 갤럭시 S2
- layout-xhdpi : 갤럭시 S3, 갤럭시 노트
- layout-xlarge-mdpi : 갤럭시 노트 10.1

Drawable 폴더 2종류 만들기

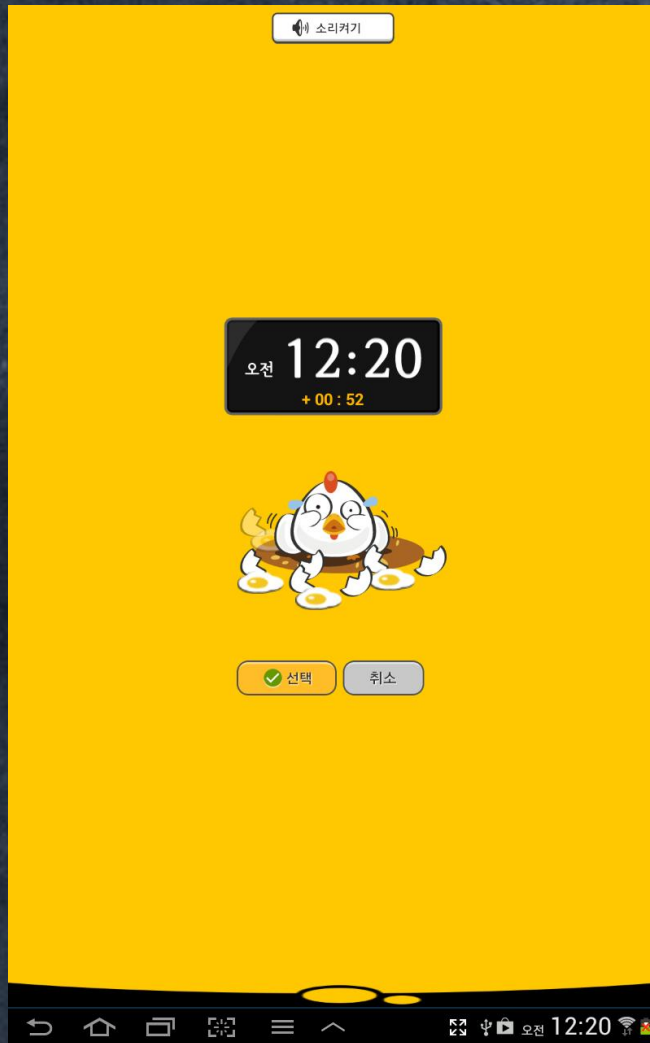
- drawable-hdpi : 800*480용 이미지
- drawable-xhdpi : 1280*800, 1280*720용 이미지

통신사별 안드로이드 단말기 점유율표(알람몬, 사용자 기준)

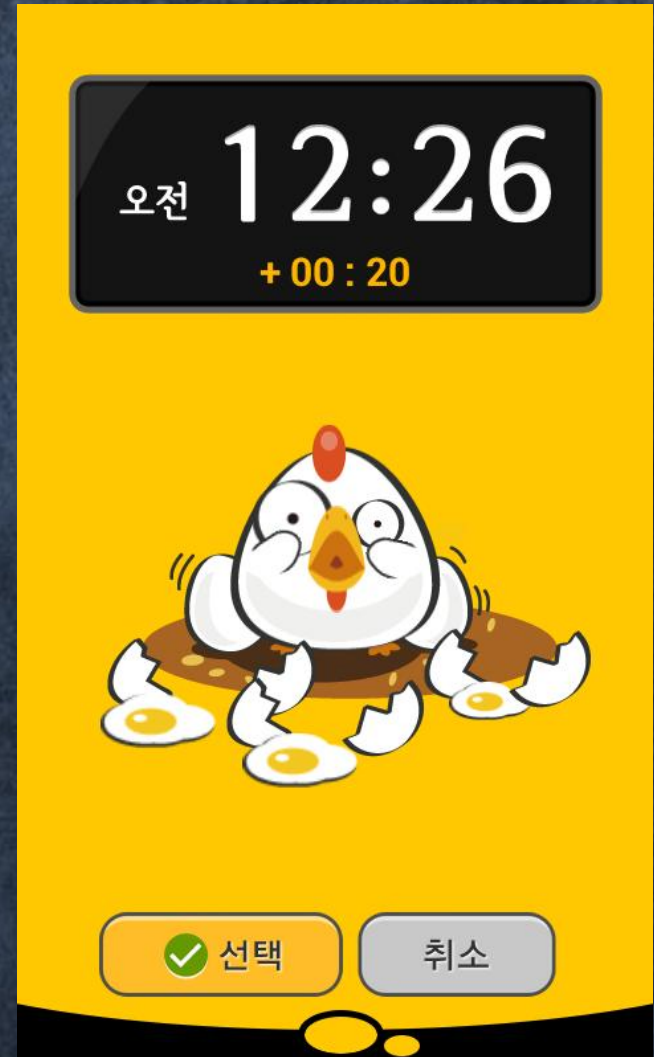
(2012.05 ~ 2012.06 기준)

Sk Telecom. Ltd.			Korea Telecom			Lg Telecom		
Model	Rank(%)	Value	Model	Rank(%)	Value	Model	Rank(%)	Value
Samsung Galaxy S II	1	862,910	Samsung Galaxy S	1	396,373	Samsung Galaxy Note	1	366,287
Samsung Galaxy Note	2	559,445	Samsung Galaxy Note	2	342,852	LG Optimus LTE	2	241,404
Samsung Galaxy S	3	419,947	Pantech Vega Racer	3	194,282	Samsung Galaxy S II	3	152,155
Samsung Galaxy S II LTE	4	328,340	Samsung Galaxy S II	4	129,507	Samsung Galaxy S II HD	4	136,372
Pantech Vega Racer	5	230,154	Nexus S	5	118,568	Pantech Vega LTE EX	5	104,001
Pantech Sky Vega	6	154,951	Samsung Galaxy Nexus	6	108,650	LG-F100L	6	87,705
LG Optimus LTE	7	83,179	SHV-E120K	7	56,759	Pantech Vega Racer	7	77,165
Samsung Galaxy Nexus	8	73,046	HTC EVO 4G+	8	51,545	LG Optimus Big	8	60,393

1. 기기 화면 크기에 따른 문제!!



갤럭시 탭 10.1인치



갤럭시 S2 4.3인치

COCOS 2D 엔진으로 자동 스케일링



갤럭시 탭 10.1인치



갤럭시 S2 4.3인치

2. 기기 해상도에 따른 문제!!

COCOS2D 게임 엔진을 이용하여 자동 스케일링 후 나머지 부분 Crop



갤럭시 노트
16:10 (1280*800)



갤럭시 S3
16:9 (1280*720)



옵티머스 뷰
4:3 (1024*768)

Cocos2D 사용 시...

- Cocos2D로 UI를 나타낼 View 설정
- Handler 객체 전달하여 Activity와 통신
- **Activity Context** 전달
 - String Resource 가져오기
 - 진동, 전화 등과 같이 안드로이드 시스템 서비스
사용시

3. 언어별 글자 길이 문제

한국어 : 배고픈 양에게 먹이를 주세요!

중국어 : 主人，放飯的時間到了!

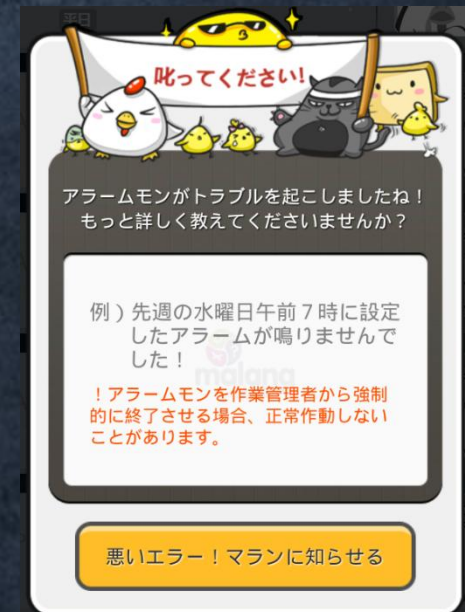
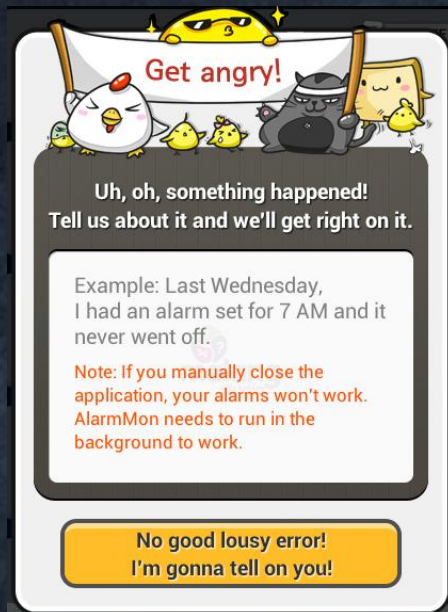
일본어 : たいへん、ひつじが腹ペコ!

餌を与えて

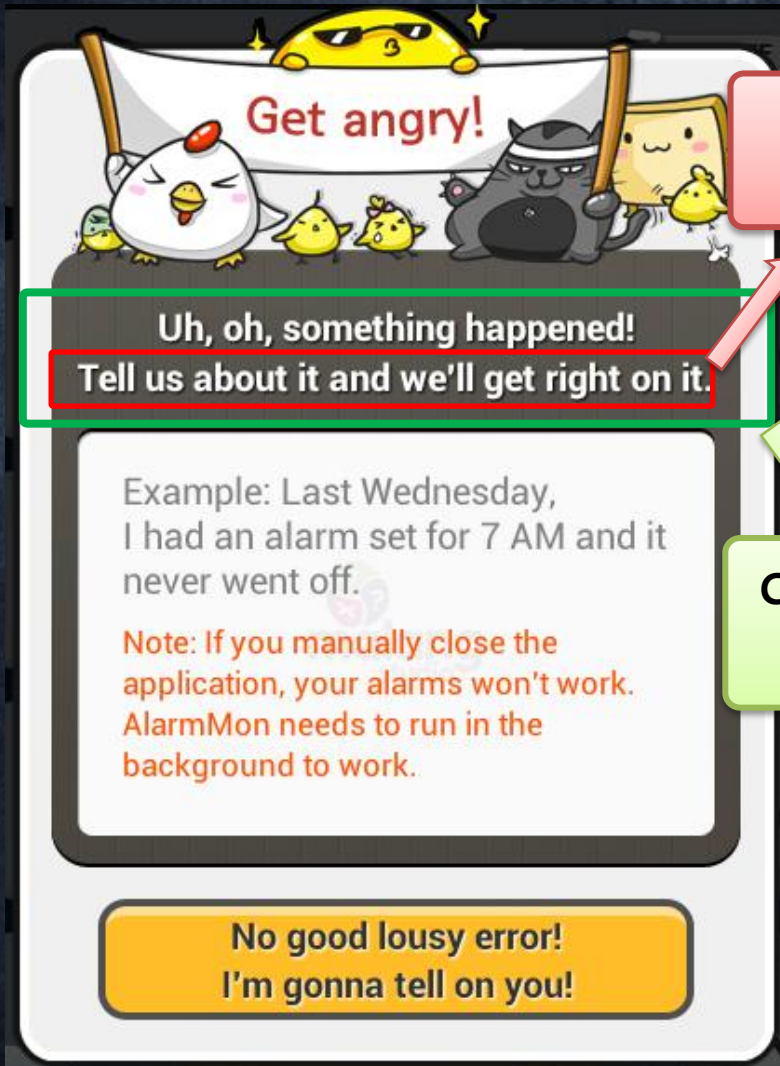
영어 : Feed the hungry sheep !

Custom TextView

- 글자 길이가 지정된 길이를 초과한다면 지정된 길이를 초과하지 않도록 폰트 사이즈 조절
- 문자열에 개행(\n) 문자가 있다면 문자열들을 개행 문자 기준으로 나눈뒤 가장 긴 부분 기준으로 폰트 사이즈 조절
- 설정된 텍스트 뷰 길이 기준으로 원하는 비율이 문자열의 최대 길이가 될수 있도록 폰트 사이즈 조절

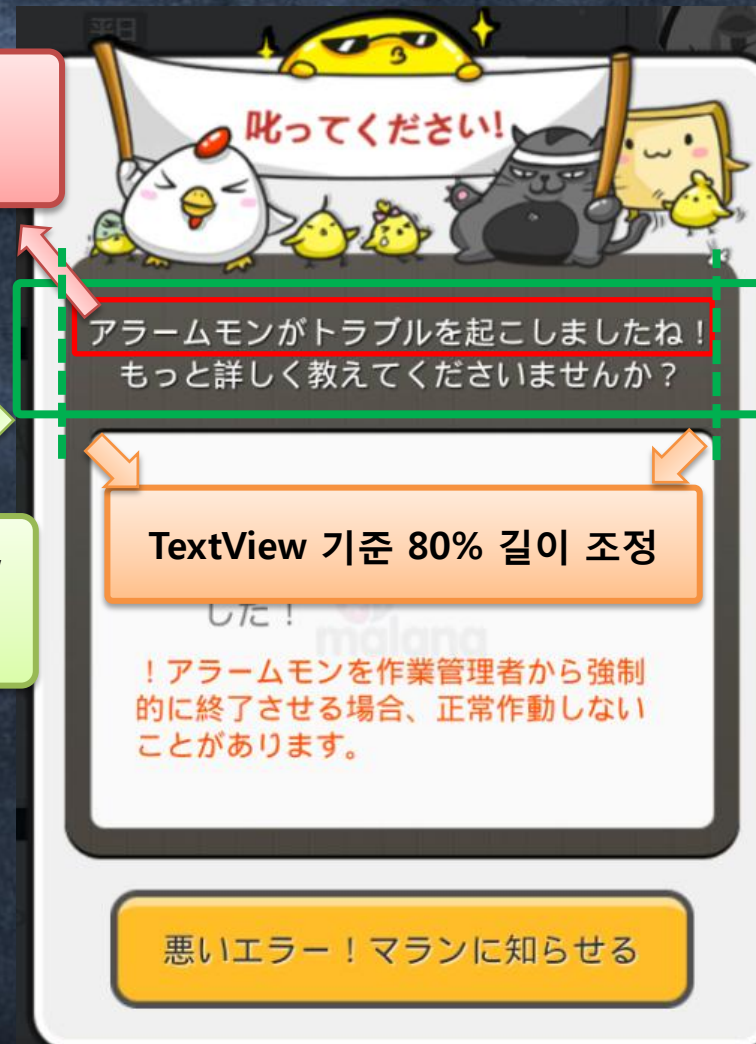


Custom TextView



개행 기준
가장 긴 문자열

Custom TextView
원래 크기



```
<RelativeLayout                android:layout_width= " wrap_content"
                                android:layout_height= "wrap_content" >
```

```
<ImageView    android:id= "@+id/imageView1"
                android:src= "@drawable/ah_send_error2"
                android:layout_width= "wrap_content"
                android:layout_height= "wrap_content"/>
```

```
<com.malangstudio.alarmmon.util.ResizeTextView
```

```
    android:layout_width= "match_parent"
    android:layout_height= "match_parent"    android:gravity= "center"
    android:layout_alignBottom= "@id/imageView1"
    android:layout_alignLeft= "@id/imageView1"
    android:layout_alignRight= "@id/imageView1"
    android:layout_alignTop= "@id/imageView1"
    minTextSize= "10"    android:textSize= "15sp"
    widthPercent= "80"    android:text= "@string/SendError_Title"/>
```

```
</RelativeLayout>
```



```
public class ResizeTextView extends TextView
{
    private Paint textPaint;
    private float preferredTextSize, minTextSize;
    private int widthPercent = 100;

    public ResizeTextView(Context context)
    {
        this(context, null);
    }

    public ResizeTextView(Context context, AttributeSet attrs)
    {
        this(context, attrs, 0);
    }

    // xml로 정의
    public ResizeTextView(Context context, AttributeSet attrs, int defStyle)
    {
        super(context, attrs, defStyle);
        this.textPaint = new Paint();
        this.widthPercent = Integer.parseInt(attrs.getAttributeValue(null, "widthPercent"));
        this.minTextSize = Float.parseFloat(attrs.getAttributeValue(null, "minTextSize"));
        this.preferredTextSize = this.getTextSize();
    }

    @Override
    protected void onTextChanged(final CharSequence text, final int start, final int before, final int after)
    {
        this.refitText(text.toString(), this.getWidth());
    }

    @Override
    protected void onSizeChanged(int width, int height, int oldwidth, int oldheight)
    {
        if (width != oldwidth)
            this.refitText(this.getText().toString(), this.getWidth());
    }
}
```

```

private void refitText(String text, int textWidth){
    if (widthPercent != 100)
        textWidth = this.getWidth() * widthPercent / 100;

    this.textPaint.set(this.getPaint());

    float max = 0;
    String[] array;
    array = text.split("\n");
    for (int i = 0; i < array.length; i++){
        if (max < this.textPaint.measureText(array[i])){
            text = array[i];
            max = this.textPaint.measureText(array[i]);
        }
    }

    if (textWidth <= 0 || text == null || text.length() == 0)
        return;
    int targetWidth = textWidth - this.getPaddingLeft() - this.getPaddingRight();
    final float threshold = 0.5f; // 한계값

    while ((this.preferredTextSize - this.minTextSize) > threshold){
        float size = (this.preferredTextSize + this.minTextSize) / 2;
        this.textPaint.setTextSize(size);
        if (this.textPaint.measureText(text) >= targetWidth)
            this.preferredTextSize = size; // too big
        else
            this.minTextSize = size; // too small
    }
    this.setTextSize(TypedValue.COMPLEX_UNIT_PX, this.minTextSize);
}

```

원하는 가로 비율만큼만
문자열 배치

개행문자 기준으로
가장 긴 문자열 찾기

이진 탐색을 통해서
최적의 Font Size 계산

Part 2. Network

UI

UI Pattern

UI Binding

Actionbar

Hybrid

Network

(Un)Marshaller

(Un)Marshller + Dispatcher

SNS

Common

Logging

DI

Lightweight DI

BaaS

User Behavior

Logging

Notification

Open Source Bas
s

Standalone

VS

Online

앱으로 돈을 벌기 위해서는...

72%

OF REVENUE
COMES FROM APPS
FEATURING IN-APP
PURCHASES



source : Distimo

네트워크 통신은 필수!!!

일반적인 네트워크 통신

1. 다운로드

1. 네트워크 통신
2. 데이터를 객체로 언마샬링

2. 업로드

1. 객체를 데이터로 마샬링
2. 네트워크 통신

그래서, 코드는?

```
HttpClient client = new DefaultHttpClient();
HttpGet get = new HttpGet();
get.setURI(new URI("http://www.google.com/"));
HttpResponse resp = client.execute(get);
BufferedReader br = new BufferedReader(new
InputStreamReader(resp.getEntity().getContent()));
String str = null;
StringBuffer sb = new StringBuffer();
while ((str = br.readLine()) != null) {
    sb.append(str).append(" ");
}
br.close();
```


그래서, 코드는?

```
<categoryList>
  <categorys>
    <category>
      <id>1</id>
      <name>AA</name>
      <date>2011-11-01</date>
    </category>
    <category>
      <id>2</id>
      <name>BB</name>
      <date>2011-12-10</date>
    </category>
  </categorys>
</categoryList>
```

그래서, 코드는?

```
while(true) {  
    int eventType = xpp.nextTag();  
    if(eventType == XmlPullParser.START_TAG) {  
        String tag = xpp.getName();  
        if("id".equals(tag)) {  
            category.setId(Integer.parseInt(xpp.nextText()));  
        }else if ("name".equals(tag)) {  
            category.setName(xpp.nextText());  
        }else if ("date".equals(tag)) {  
            category.setDate(xpp.nextText());  
        }else{  
            if(XmlPullParser.TEXT == xpp.next())  
                xpp.nextTag();  
        }  
    }  
}
```

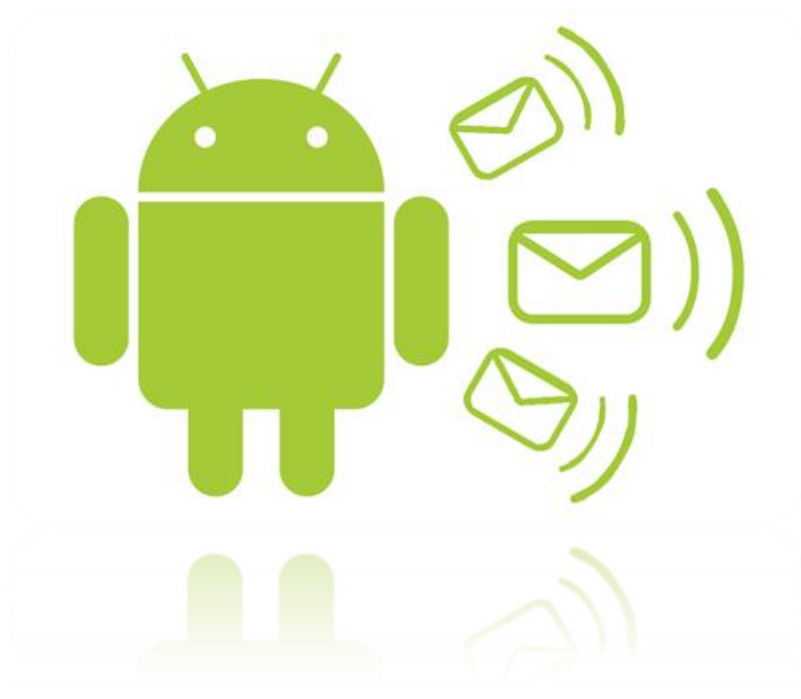

네트워크 작업은!

불편한 데이터
마샬링/언마샬링

반복적인
네트워크 통신

5장.

데이터 마샬링/언마샬링



같이 뭉쳐 다니는 파라미터는 ParameterObject로 묶어라!

BeforeRefactoringCustomer

+AmountInvoicedIn(start: Date, end: Date)
+AmountRecievedIn(start: Date, end: Date)
+AmountOverDueIn(start: Date, end: Date)

AfterRefactoringCustomer

+AmountInvoicedIn(Period: DateRange)
+AmountRecievedIn(Period: DateRange)
+AmountOverDueIn(Period: DateRange)

Simple Framework를
이용하세요.

<http://simple.sourceforge.net/>

이렇게 됩니다.

```
<categoryList>
  <category>
    <category>
      <id>1</id>
      <name>fruit</name>
      <date>201110051200</date>
    </category>
    <category>
      <id>2</id>
      <name>animal</name>
      <date>201110051200</date>
    </category>
    <category>
      <id>3</id>
      <name>food</name>
      <date>201110051200</date>
    </category>
  </category>
</categoryList>
```

@Root

```
public class CategoryList {
```

@ElementList

```
private ArrayList<Category> categorys;
```

```
public void setCategorys(ArrayList<Category> categorys) {
```

```
    this.categorys = categorys;
```

```
}
```

```
public ArrayList<Category> getCategorys() {
    return categorys;
```

```
}
```

```
}
```

이렇게 됩니다.

```
while(true) {
    int eventType = xpp.nextTag();
    if(eventType == XmlPullParser.START_TAG) {
        String tag = xpp.getName();
        if("id".equals(tag)) {
            category.setId(Integer.parseInt(xpp.nextText()));
        }
        Serializer serializer = new Persister();
        Reader reader = new StringReader(xmlData);
        CategoryList categoryList =
            serializer.read(CategoryList.class, reader, false);
        category.setDate(xpp.nextText());
    }else{
        if(XmlPullParser.TEXT == xpp.next())
            xpp.nextTag();
    }
}
else
    break;
}
```


Gson를
이용하세요.

<http://code.google.com/p/google-gson/>

이렇게 됩니다.

```
{
  "firstName": "John",
  "lastName": "Smith",
  "address": {
    "street": "100 Main St",
    "city": "New York",
    "state": "NY",
    "zip": "10001"
  },
  "phoneNumber": "646 123-4567"
}
```

```
try {
  Object obj = JSONValue.parseWithException(s);
  JSONArray array = (JSONArray)obj;

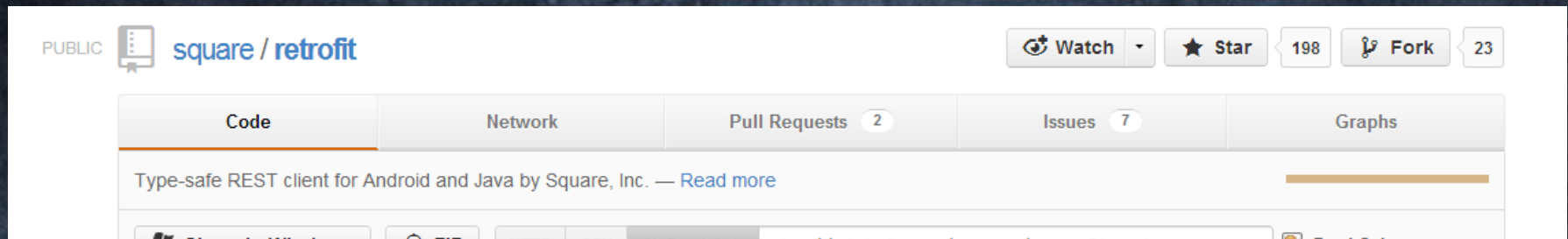
  Data data = new
  Gson().fromJson(json, Data.class);
}
```


6장. + Dispatcher

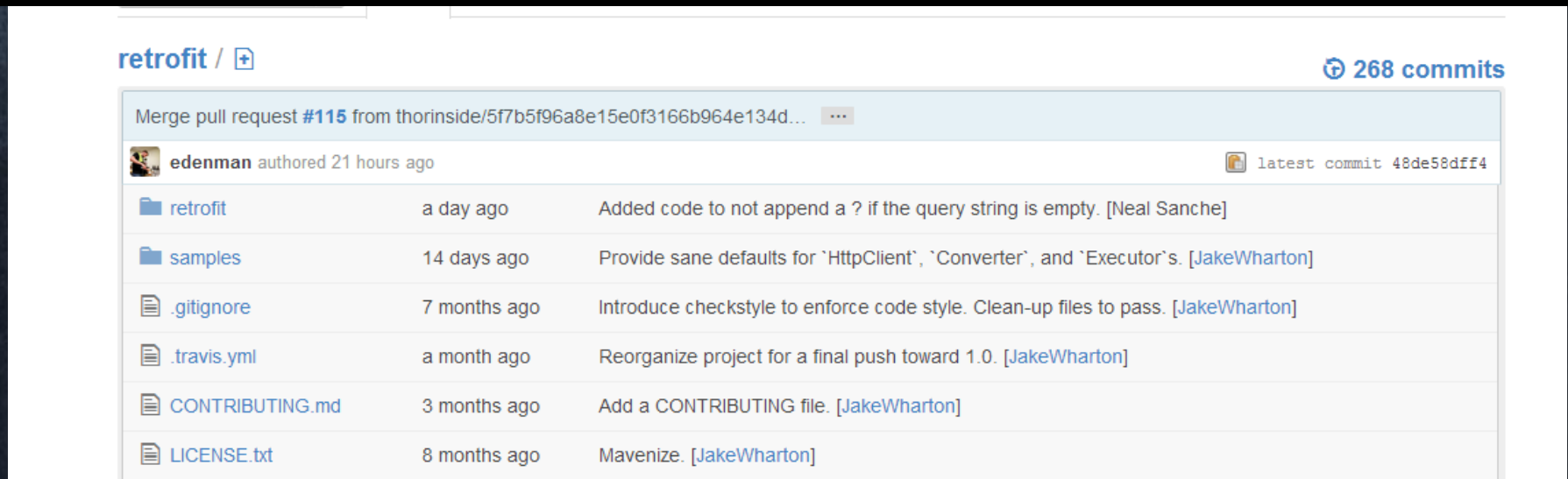


반복적인 네트워크 통신

retrofit



<https://github.com/square/retrofit>



샘플 어플리케이션!

Github user viewer!

Github api 를 사용하여 정보를 출력

- 사용자 검색
- 사용자의 followers
- 사용자의 following



Users API

- [Get a single user](#)
- [Get the authenticated user](#)
- [Update the authenticated user](#)
- [Get all users](#)

Many of the resources on the users API provide a shortcut for getting information about the currently authenticated user. If a request URL does not include a `:user` parameter then the

[Overview](#)[Activity](#)[Gists](#)[Git Data](#)[Issues](#)

http://developer.github.com/v3/users

GET /users/:user

Response

Status: 200 OK
X-RateLimit-Limit: 5000
X-RateLimit-Remaining: 4999

```
{
  "login": "octocat",
  "id": 1,
  "avatar_url": "https://github.com/images/error/octocat_happy.gif",
  "gravatar_id": "somehexcode",
  "url": "https://api.github.com/users/octocat",
  "name": "monalisa octocat",
  "company": "GitHub",
  "blog": "https://github.com/blog",
  "location": "San Francisco",
  "email": "octocat@github.com",
```

[Pull Requests](#)[Repositories](#)[Users](#)[Emails](#)[Followers](#)[Keys](#)[Search](#)[Gitignore](#)[Markdown](#)

사용자 정보 조회 API

https://api.github.com/users/{user_id}

```
1 {
2   "created_at": "2011-09-28T08:21:50Z",
3   "type": "User",
4   "followers": 3,
5   "avatar_url": "https://secure.gravatar.com/avatar/b150400a18767d0c9b0c24672bc3204f?
d=https://a248.e.akamai.net/assets.github.com%2Fimages%2Fgravatars%2Fgravatar-user-420.png",
6   "public_gists": 0,
7   "gravatar_id": "b150400a18767d0c9b0c24672bc3204f",
8   "public_repos": 7,
9   "html_url": "https://github.com/realbeast",
10  "url": "https://api.github.com/users/realbeast",
11  "gists_url": "https://api.github.com/users/realbeast/gists{/gist_id}",
12  "email": "moleskine7@gmail.com",
13  "events_url": "https://api.github.com/users/realbeast/events{/privacy}",
14  "location": "Seoul",
15  "repos_url": "https://api.github.com/users/realbeast/repos",
16  "organizations_url": "https://api.github.com/users/realbeast/orgs",
17  "followers_url": "https://api.github.com/users/realbeast/followers",
18  "name": "Sung-ju Jin",
19  "hireable": false,
20  "received_events_url": "https://api.github.com/users/realbeast/received_events",
21  "starred_url": "https://api.github.com/users/realbeast/starred{/owner}/{/repo}",
22  "following_url": "https://api.github.com/users/realbeast/following",
23  "following": 4,
24  "company": null,
25  "blog": "http://blog.softwaregeeks.org",
26  "id": 1086275,
27  "login": "realbeast",
28  "bio": null,
29  "subscriptions_url": "https://api.github.com/users/realbeast/subscriptions"
30 }
```


사용자 클래스

```
class User {  
    String id;  
    String login;  
    String name;  
    String blog;  
    String avatar_url;  
}
```

통신할 인터페이스 정의

```
interface GitHub {  
    @GET("users/{id}")  
    User getUser(@Named("id")  
                  String id);  
}
```


실제 통신

```
RestAdapter restAdapter = new  
RestAdapter.Builder()  
.setServer(new Server("https://api.github.com/"))  
.setClient(new DefaultHttpClient())  
.setConverter(new GsonConverter(new Gson()))  
.build();
```

```
GitHub github =  
restAdapter.create(GitHub.class);  
User me =  
github.getUser("realbeast");
```

통신할 인터페이스 추가!!!

```
interface GitHub {  
    @GET("users/{id}")  
    User getUser(@Named("id") String id);  
  
    @GET("users/{id}/followers")  
    List<User> getFollowers(@Named("id")  
    String id);  
  
    @GET("users/{id}/following")  
    List<User> getFollowing(@Named("id")  
    String id);  
}
```


추가한 인터페이스 통신

```
RestAdapter restAdapter = new  
RestAdapter.Builder()  
.setServer(new Server("https://api.github.com/"))  
.setClient(new DefaultHttpClient())  
.setConverter(new GsonConverter(new Gson()))  
.build();
```

```
List<User> followers =  
github.getFollowers("realbeast");  
for(User user : followers) {  
    System.out.println(user.login);  
}
```

```
List<User> following =  
github.getFollowing("realbeast");  
for(User user : following) {  
    System.out.println(user.login);  
}
```

7장. SNS Layer

facebook framework

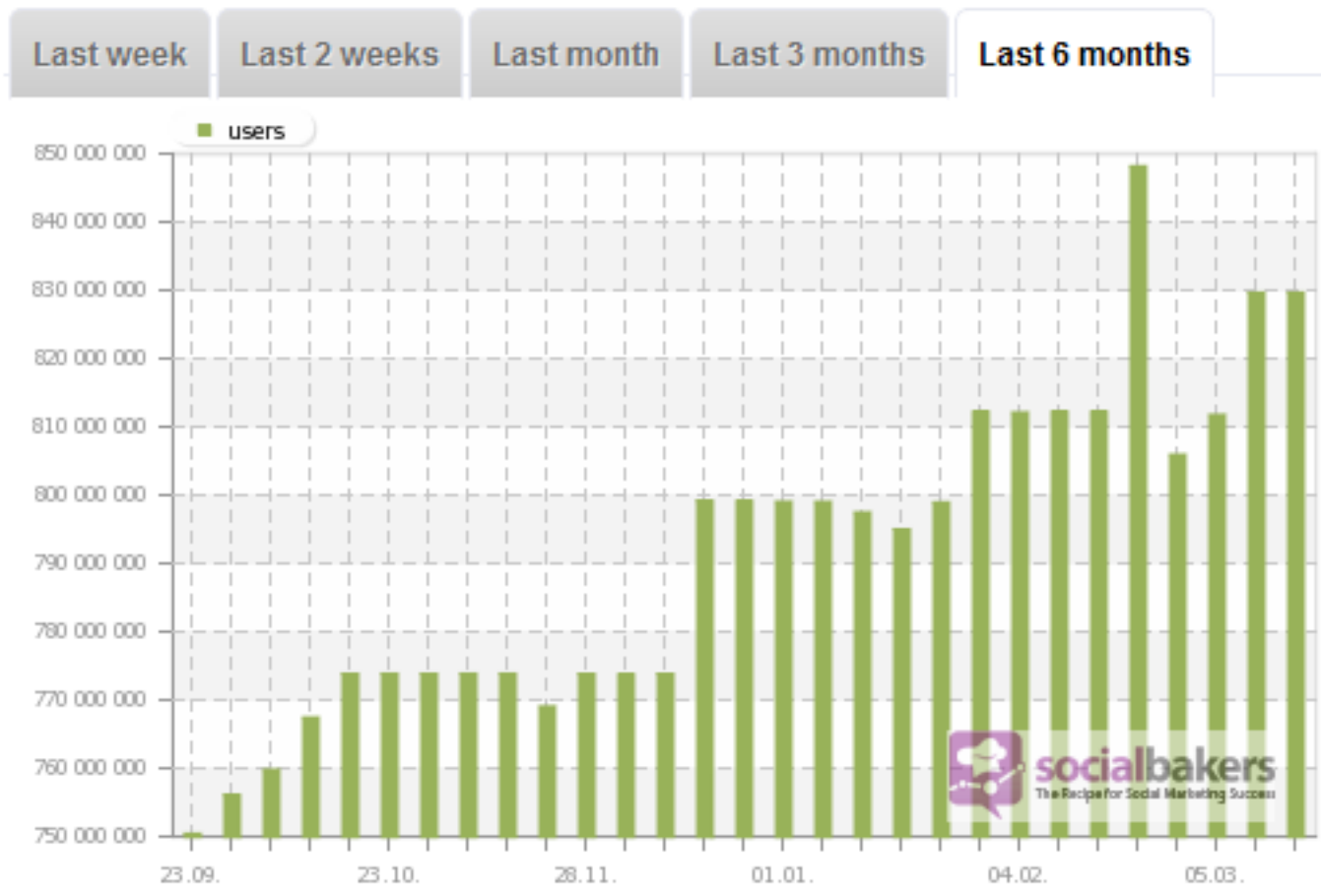


facebook

세계에서 가장 큰 서비스

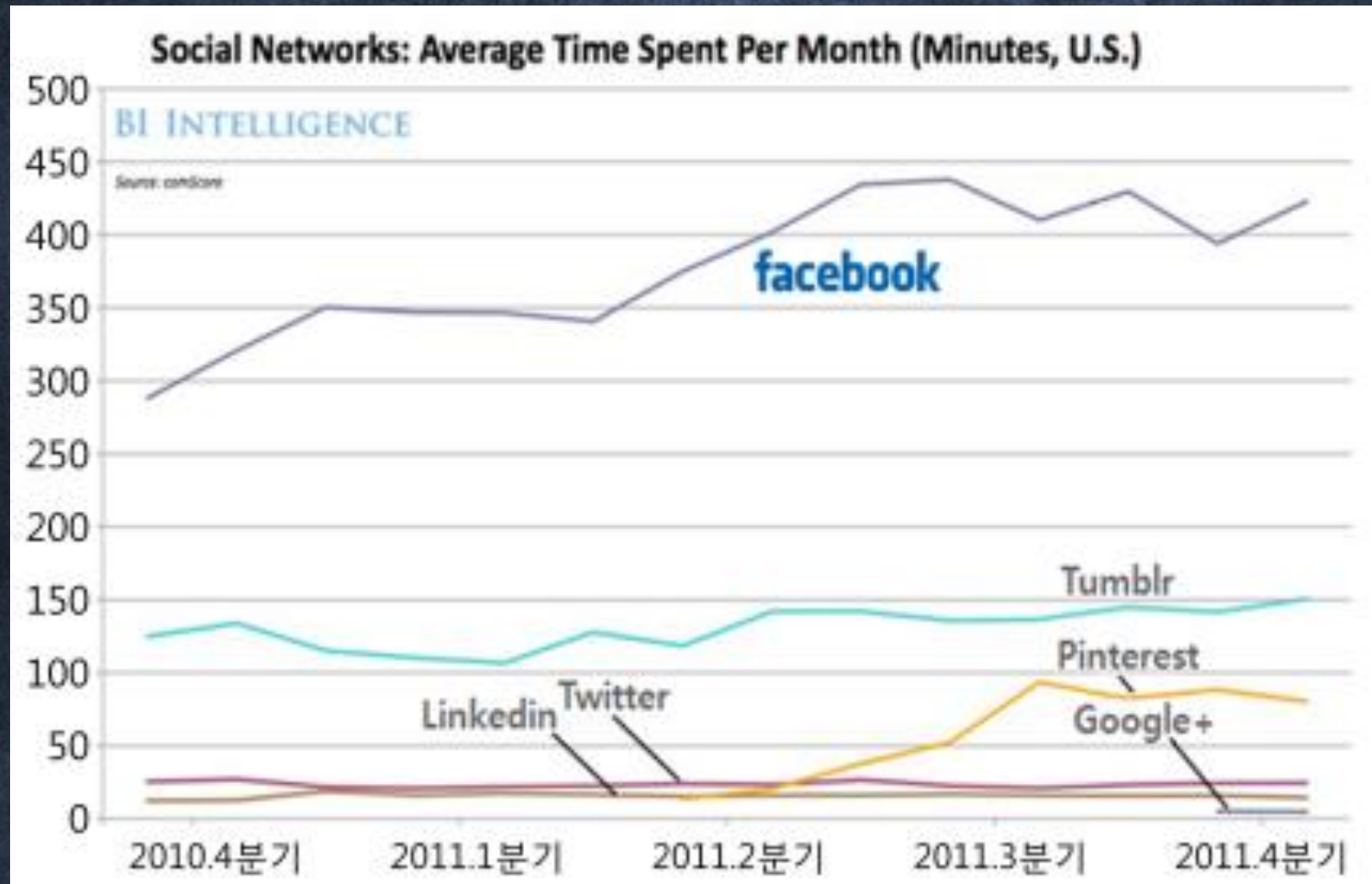
(약 8억5천만)

Worldwide Facebook Users **854,750,780**



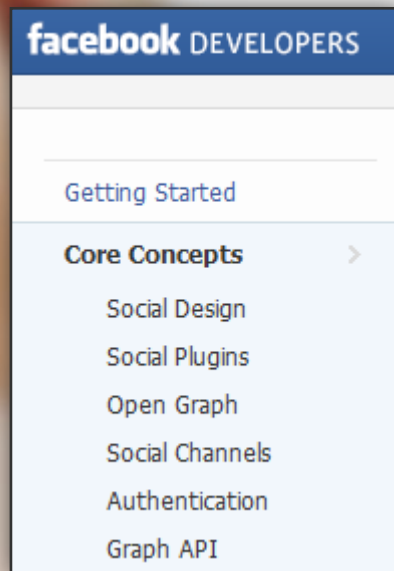


가장 오랜시간 이용하는 서비스



facebook 을 개발하려면...

너무 많은 것을 알아야 한다.



Social Plugin ?
Open Graph ?
Social Channels ?
Authentication ?
Graph API ?

그래서 우리는..

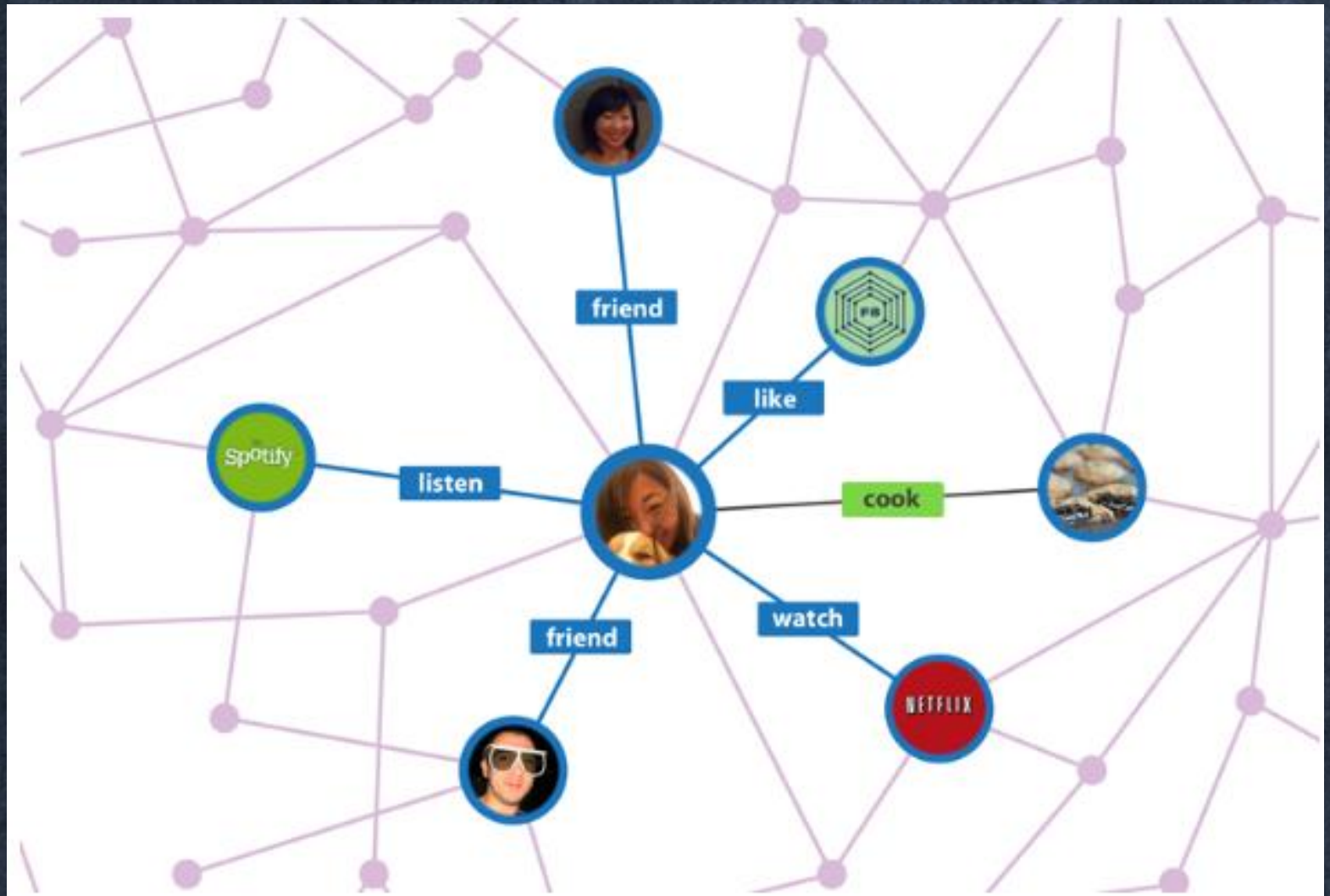
facebook framework



facebook framework Project

를 시작했습니다.

Facebook 기본 개념 - 소셜 그래프



About Graph API

- Graph API
 - 소셜 그래프의 Object(친구, 페이지, 사진 등)를 다루는 API
- Graph API의 구조
 - https://graph.facebook.com/OBJECT_ID/CONNECTION_TYPE
- ID는 사용자, 페이지, 이벤트, 사진 등의 Object ID
 - 약 20개의 Object를 지원
 - 모든 Object의 ID는 unique하다
 - JSON 형태로 응답을 받는다

Graph API Example

- <https://graph.facebook.com/100001066448386/>

```
{
  "id": "100001066448386",
  "name": "신재명",
  "first_name": "재명",
  "last_name": "신",
  "link": "https://www.facebook.com/JaemyungShin",
  "username": "JaemyungShin",
  "gender": "male",
  "locale": "ko_KR"
}
```

신재명 ID

- <https://graph.facebook.com/40796308305/>

```
{
  "id": "40796308305",
  "name": "Coca-Cola",
  "picture": "http://profile.ak.fbcdn.net/hprofile-ak-snc4/174560_4079",
  "link": "https://www.facebook.com/coca-cola",
  "likes": 41747534,
  "cover": {
    "cover_id": "10150682306963306",
    "source": "http://a7.sphotos.ak.fbcdn.net/hphotos-ak-snc7/s720x72",
    "offset_y": 0
  }
}
```

Graph API - Connection

- Graph API의 구조
 - https://graph.facebook.com/OBJECT_ID/CONNECTION_TYPE
- Connection이란 ?
 - Object의 연관(관계) 개념
- User object의 Connection 종류
 - Family, friends, album, likes, posts ...
 - User object 경우 약 25개의 Connection 을 제공
 - <https://developers.facebook.com/docs/reference/api/user/> 참고

Connection - Example

- Graph API 예시
 - <https://graph.facebook.com/100001066448386/friends>

```
{
  "data": [
    {
      "name": "Sungmoo Ban",
      "id": "0000000000"
    },
    {
      "name": "Grace HyeYoun Yang",
      "id": "0000000000"
    },
    {
      "name": "Sanghyun Mark Kim",
```

그럼 보안문제는 ???

Access Token

- 페이스북 API를 사용하기 위해 Access Token(인증)이 필요하다.
- Graph API Explorer를 통하여 쉽게 Access Token을 받아올 수 있음

About Graph API

Select Permissions



User Data Permissions

Friends Data Permissions

Extended Permissions

- | | | |
|---|--|---|
| <input checked="" type="checkbox"/> user_about_me | <input checked="" type="checkbox"/> user_activities | <input checked="" type="checkbox"/> user_birthday |
| <input checked="" type="checkbox"/> user_checkins | <input checked="" type="checkbox"/> user_education_history | <input checked="" type="checkbox"/> user_events |
| <input checked="" type="checkbox"/> user_games_activity | <input checked="" type="checkbox"/> user_groups | <input checked="" type="checkbox"/> user_hometown |
| <input checked="" type="checkbox"/> user_interests | <input checked="" type="checkbox"/> user_likes | <input checked="" type="checkbox"/> user_location |
| <input checked="" type="checkbox"/> user_notes | <input checked="" type="checkbox"/> user_online_presence | <input checked="" type="checkbox"/> user_photo_video_tags |
| <input checked="" type="checkbox"/> user_photos | <input checked="" type="checkbox"/> user_questions | <input checked="" type="checkbox"/> user_relationship_details |
| <input checked="" type="checkbox"/> user_relationships | <input checked="" type="checkbox"/> user_religion_politics | <input checked="" type="checkbox"/> user_status |
| <input checked="" type="checkbox"/> user_subscriptions | <input checked="" type="checkbox"/> user_videos | <input checked="" type="checkbox"/> user_website |
| <input checked="" type="checkbox"/> user_work_history | | |

Basic Permissions already included by default

Get Access Token

취소

<https://developers.facebook.com/tools/explorer#!/tools/explorer>

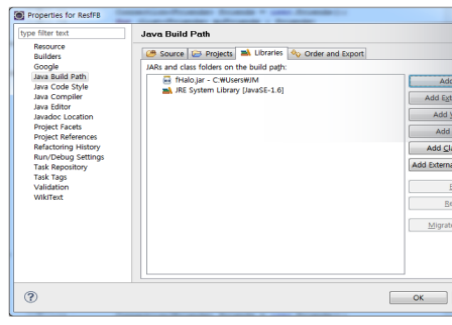
사용법

Step1 Import fHalo.jar

Step 1: Import fHalo.jar

To use fHalo, you have to import the fHalo.jar.

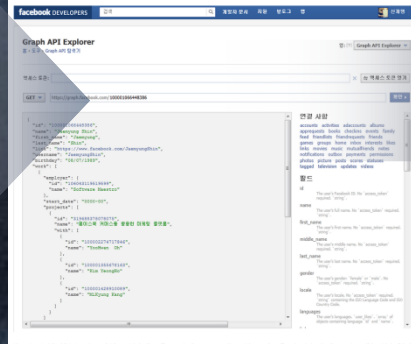
right click the project -> properties -> add External JARs -> add fHalo.jar



Step2 Get access token

Step 2: Get access token

It's really easy with the Facebook Graph API. Sign in to Facebook and navigate to the Graph API Explorer app. Click the "Get Access Token" button. That's it!



Step3 Create instance

Step 3: Create instance

In Facebook Graph API, there are lots of objects(e.g., people, photos, events, and pages) and the connection between them(e.g., relationships, shared content, and photo tags).

fHalo provide object and connection type. To get specific object information, you have to create instance. If you want to get particular page, create user instance or page instance like below code.

```
//To get user information(get user's information) - "me" means my id
User user = new User();
user = user.createInstance("me");

//To get console page information(get page's information)
Page page = new Page();
page = page.createInstance("facebook");
```

Step4 Using API

Step 4: Using API

You can run some sample code, jump straight to the fHalo javadoc, or learn by example below.

1. Get object information (e.g., user, page...)

```
//To get user information and page information(user and page)
//To get user information
User user = new User();
user = user.createInstance("me"); // "me" means my id
//To get console page information
Page page = new Page();
page = page.createInstance("facebook");
```

2. Get connection information (e.g., friends, feed...)

```
//To get friends list (get connection)
//Initialization
User user = new User();
user = user.createInstance("me"); // "me" means my id

//Get connection you wanted.
//fHalo provide primary connection types (e.g., Friends, Posts, Photos...)
Connections<Friends> friends = user.friends();

//Connections support paging and you can iterate
for (Friends friend : friends)
    System.out.println("My friend: " + user.getName());
```


Access Token

- 페이스북 API를 사용하기 위해 Access Token(인증)이 필요하다.
- Graph API Explorer를 통하여 쉽게 Access Token을 받아올 수 있음

친구 리스트 가져오기 (페이스북 API)

```
HttpsURLConnection connection = null;
BufferedReader bufferedReader = null;

URL url = new URL("https://graph.facebook.com/me/friends?access_token=ACCESS_TOKEN");
connection = (HttpsURLConnection) url.openConnection();

connection.setRequestMethod("GET");
connection.setRequestProperty("Content-Type" , "application/x-www-form-urlencoded");
connection.connect();

InputStreamReader reader = new InputStreamReader(connection.getInputStream());
bufferedReader = new BufferedReader(reader);

String temp = null;
StringBuffer buffer = new StringBuffer();
while ((temp = bufferedReader.readLine()) != null) {
    buffer.append(temp);
}
JsonObject json = new JsonObject(buffer.toString());
JsonArray jsonArray = json.getJSONArray("data");
for(int i=0;i<jsonArray.length();i++){
    JsonObject jsonObject = jsonArray.getJSONObject(i);
    System.out.println("My friends : " + jsonObject.get("name"));
}

connection.disconnect();
```


친구 리스트 가져오기

– Rest FB

- `Connection<User> myFriends = facebookClient.fetchConnection("me/friends", User.class);`

– fHalo

- `Connection<Friends> friends = user.friends();`

피드 올리기(페이스북 API)

```
BufferedReader bufferedReader = null;
URLConnection connection = null;

URL url = new URL("https://graph.facebook.com/me/feed?access_token=ACCESS_TOKEN");
connection = (URLConnection) url.openConnection();

connection.setDoOutput(true);
connection.setRequestMethod("POST");

String parameter = "";
parameter = URLEncoder.encode("message", "UTF-8") + "=" + URLEncoder.encode("Message Test", "UTF-8");
parameter += "&" + URLEncoder.encode("description", "UTF-8") + "=" + URLEncoder.encode("Description Test", "UTF-8");
parameter += "&" + URLEncoder.encode("caption", "UTF-8") + "=" + URLEncoder.encode("Caption Test", "UTF-8");

OutputStreamWriter outputStreamWriter = new OutputStreamWriter(connection.getOutputStream());
outputStreamWriter.write(parameter);
outputStreamWriter.flush();

bufferedReader = new BufferedReader(new InputStreamReader(connection.getInputStream(), "UTF-8"));
String line = null;
StringBuffer buffer = new StringBuffer();
while((line = bufferedReader.readLine()) != null) {
    System.out.println(line);
    buffer.append(line);
}

JSONObject json = new JSONObject(buffer.toString());
System.out.println("id : " + json.get("id"));
```


피드 올리기

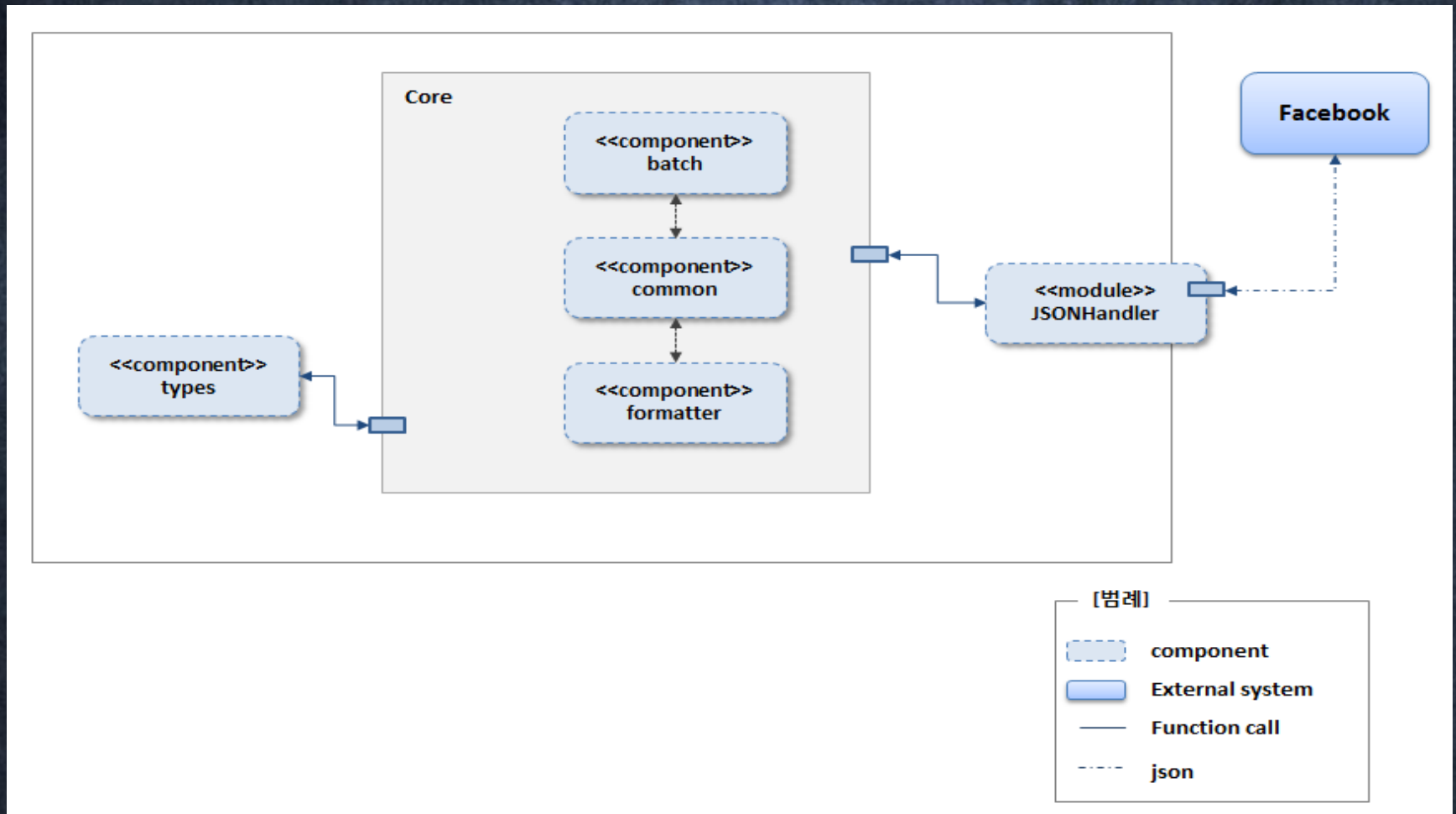
– Rest FB

- `FacebookType publishMessageResponse = facebookClient.publish("me/feed", FacebookType.class, Parameter.with("message", "RestFB test"), Parameter.with("caption", "caption test"), Parameter.with("description", "description test"),);`

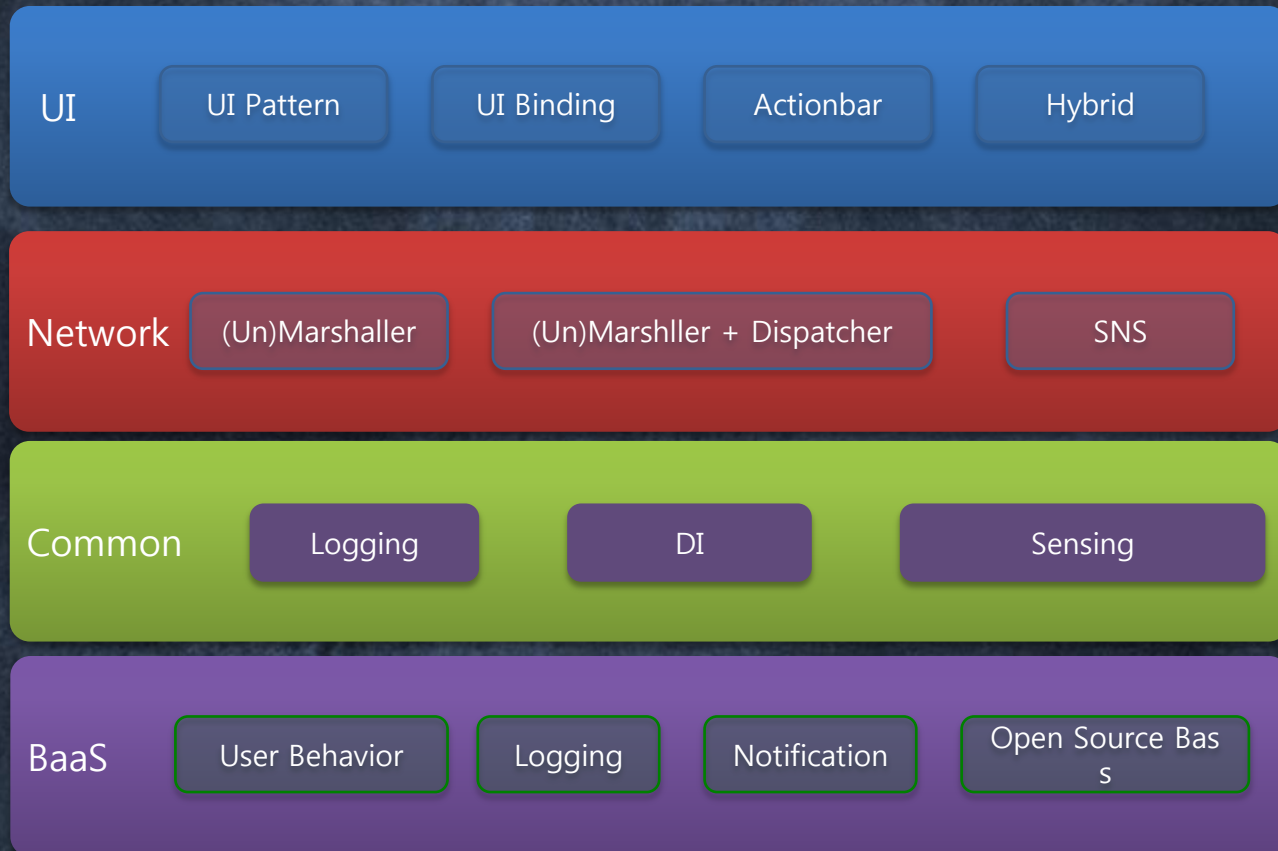
– fHalo

- `Feed feed = new feed();
feed.setMessage("Message Test");
feed.setCaption("Caption Test");
feed.setDescription("Description Test");
user.publishFeed(me, feed);`

fHalo Architecture



Part 3. Common





8장. Dependency Injection

짜증나는

안드로이드 반복적인 작업

1. UI 매핑 (findViewById)

2. 파라미터 처리

(getIntent().getExtras()...)

1. UI 매핑 (findViewById)

```
TextView subject = (TextView) findViewById(R.id.subject);
```

```
TextView writer = (TextView) findViewById(R.id.writer);
```

```
TextView date = (TextView) findViewById(R.id.date);
```

```
TextView hit = (TextView) findViewById(R.id.hit);
```

```
@ViewById
```

```
TextView subject;
```

```
@ViewById
```

```
TextView write;
```

```
@ViewById
```

```
TextView date
```

```
@ViewById
```

```
TextView hit;
```


2. 파라미터 처리

```
String id = intent.getStringExtra("id");
```

```
String name = intent.getStringExtra("name");
```

```
String nickname = intent.getStringExtra("nickname");
```

```
int sex = intent.getIntExtra("sex",0);
```

```
Object object = intent.getExtras().get("object");
```

```
@Extra("id") String id;
```

```
@Extra("name") String name;
```

```
@Extra("nickname") String nickname;
```

```
@Extra("sex") int sex;
```

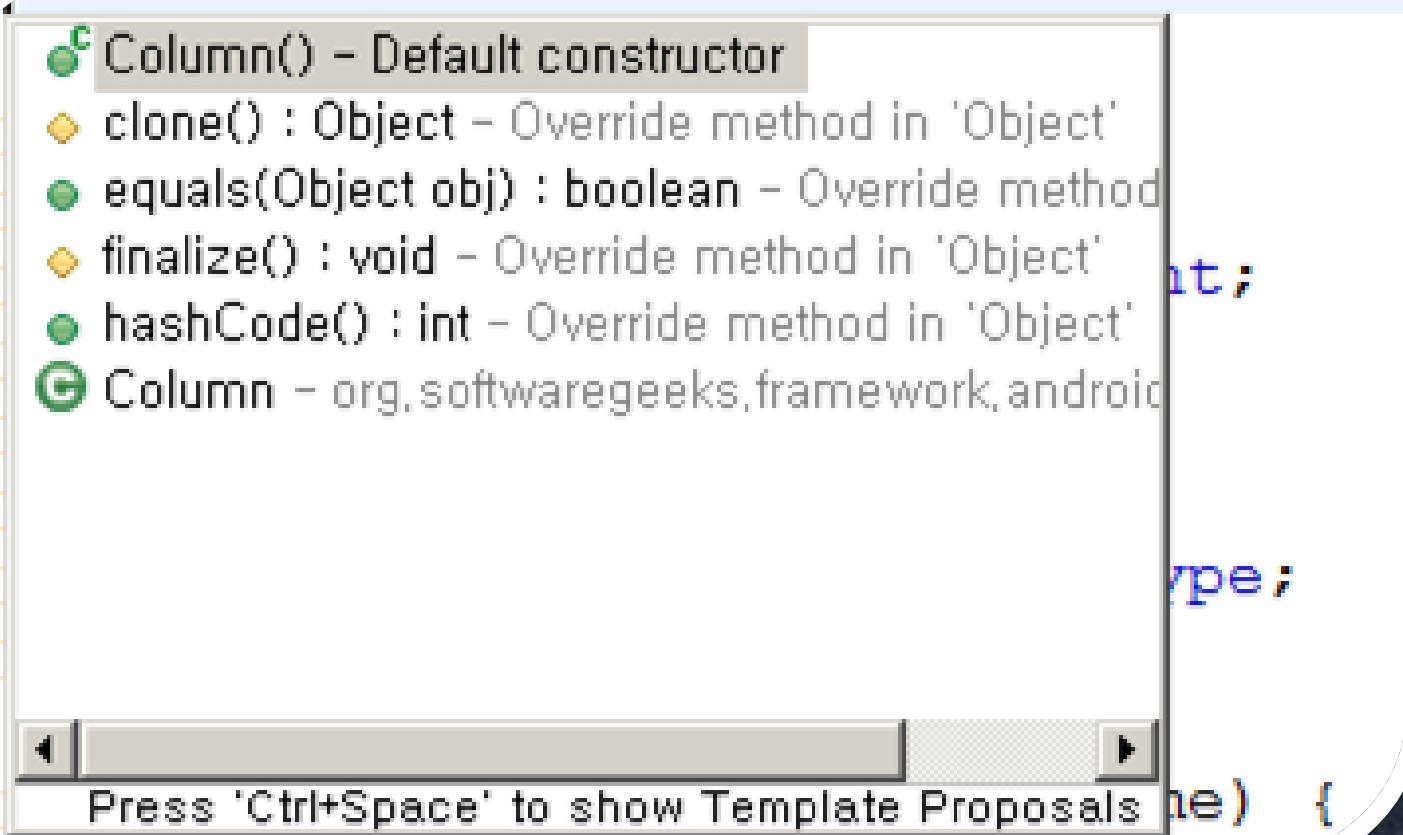
```
@Extra("object") Object object;
```

Reflection

1. (거울 등에 비친) 상
2. (빛 · 열 · 소리 등의) 반사, 반향
3. (상태 · 속성 등의) 반영

클래스 모습을 자신이 볼 수 있고, 수정할 수 도 있는
기술

이클립스 자동완성 (Ctrl+ Space)



- Column() - Default constructor
- clone() : Object - Override method in 'Object'
- equals(Object obj) : boolean - Override method
- finalize() : void - Override method in 'Object'
- hashCode() : int - Override method in 'Object'
- Column - org.softwaregeeks.framework.android

Press 'Ctrl+Space' to show Template Proposals

Annotation

1. 주석(을 달기)

```
@Retention(RetentionPolicy.RUNTIME)
```

```
public @interface Description {
```

```
    String value();
```

```
}
```

```
public class Person {
```

```
    @Description(value="이름이예요.")
```

```
    private String name;
```

```
    @Description(value="나이가예요.")
```

```
    private int age;
```


Reflection + Annotation



Google Guice on Android, version 1.1

 Search projects

Google Guice on Android Project!!

<http://code.google.com/p/roboguice/>

Project Information

★ Starred by 430 users

[Activity](#)  High
[Project feeds](#)

Code license

[Apache License 2.0](#)

Labels

guice, android, ioc, spring, java,
dependencyinjection

Members

[mbur...@gmail.com](#)
[6 contributors](#)

Featured

Wiki pages

[Downloads](#)
[Show all »](#)

Links

Groups

>> [We have an active RoboGuice Forum!](#) <<



RoboGuice smoothes out some of the wrinkles in your Android development experience and makes things simple and fun. Do you always forget to check for null when you `getIntent().getExtras()`? RoboGuice will help you. Think casting `findViewById()` to a `TextView` shouldn't be necessary? RoboGuice is on it.

RoboGuice takes the guesswork out of development. Inject your View, Resource, System Service, or any other object, and let RoboGuice take care of the details.

RoboGuice slims down your application code. Less code means fewer opportunities for bugs. It also makes your code easier to follow -- no longer is your code littered with the mechanics of the Android platform, but now it can focus on the actual logic unique to your application.

There's no "magic". Everything you need is configured explicitly for you by RoboGuice or can be overridden by you.

RoboGuice applications have been Featured on the Android Market for months. It leverages the award-winning and production-ready Google Guice library to bring you simple dependency injection lightweight enough to use in a mobile application.

It works, and people are using it. Why not give it a spin?

[Get Started!](#)



Facebook
Messenger



Pulse



Groupon



Google
Docs



OpenTable



Tript!



SwiftKey



Video to
Facebook



Cars.com



Android
Shuffle



GeoBeagle



Crossfit
Travel



Agit
(Android
Git)



Digg



TaskBomb



Wifi Syncr



CardStar



MyEpisodes
Manager



WorkTime
WorkTime



Bartab

```
class AndroidWay extends Activity {
```

```
    TextView name;
```

```
    ImageView thumbnail;
```

```
    LocationManager loc;
```

```
    Drawable icon;
```

```
    String myName;
```

```
    public void onCreate(Bundle savedInstanceState) {
```

```
        super.onCreate(savedInstanceState);
```

```
        setContentView(R.layout.main);
```

```
        name      = (TextView) findViewById(R.id.name);
```

```
        thumbnail = (ImageView) findViewById(R.id.thumbnail);
```

```
        loc       = (LocationManager) getSystemService(Activity.LOCATION
```

```
        icon      = getResources().getDrawable(R.drawable.icon);
```



```
class RoboWay extends RoboActivity {
```

```
    @InjectView(R.id.name)                TextView name;
```

```
    @InjectView(R.id.thumbnail)           ImageView thumbnail;
```

```
    @InjectResource(R.drawable.icon)      Drawable icon;
```

```
    @InjectResource(R.string.app_name)    String myName;
```

```
    @Inject                               LocationManager loc;
```

```
    public void onCreate(Bundle savedInstanceState) {
```

```
        super.onCreate(savedInstanceState);
```

```
        setContentView(R.layout.main);
```

```
        name.setText( "Hello, " + myName );
```

```
    }
```

```
}
```

RoboGuice 기능

- 기본 클래스 Inject 기능

Context

- [Context](#)
- [Application](#)
- [Activity](#)

Android Services

- [Shared Preferences](#)
- [Content Resolver](#)
- [Asset Manager](#)
- [Resources](#)
- [Location Manager](#)

- [Window Manager](#)
- [Layout Inflater](#)
- [Activity Manager](#)
- [Power Manager](#)
- [Alarm Manager](#)
- [Notification Manager](#)
- [Keyguard Manager](#)
- [Search Manager](#)
- [Vibrator Manager](#)
- [Connectivity Manager](#)
- [Wifi Manager](#)
- [Input Method Manager](#)
- [Sensor Manager](#)

RoboGuice 기능

- RoboAsyncTask

기본적인 AsyncTask를 확장하고 onException(), onFinally() 통해서 에러처리 가능

- Event handlers

OnActivityResultEvent , OnConfigurationChangedEvent, OnContentChangedEvent, OnContentViewAvailableEvent

OnCreateEvent, OnDestroyEvent 등 많은 이벤트를 받을 수 있도록 어노테이션 지원

RoboGuice 장단점

장점

1. Dependency Injection Framework 인 Google Guice를 Android 에서 사용할 수 있다.
2. 다양한 어노테이션과 기본 클래스들을 사용하여 코드를 줄일 수 있음

단점

1. 라이브러리 용량문제(guice-2.0-no_aop + roboGuice1.1.2 = 533KB)
Making Your App Smaller - <http://code.google.com/p/roboGuice/wiki/ProGuard>
2. 런타임 리플렉션 사용으로 인한 성능저하

Android

- 배터리문제
- GC(Gabage Collection)
- 느린 Reflection (특히 예전 디바이스)
- Lifecycle 관리

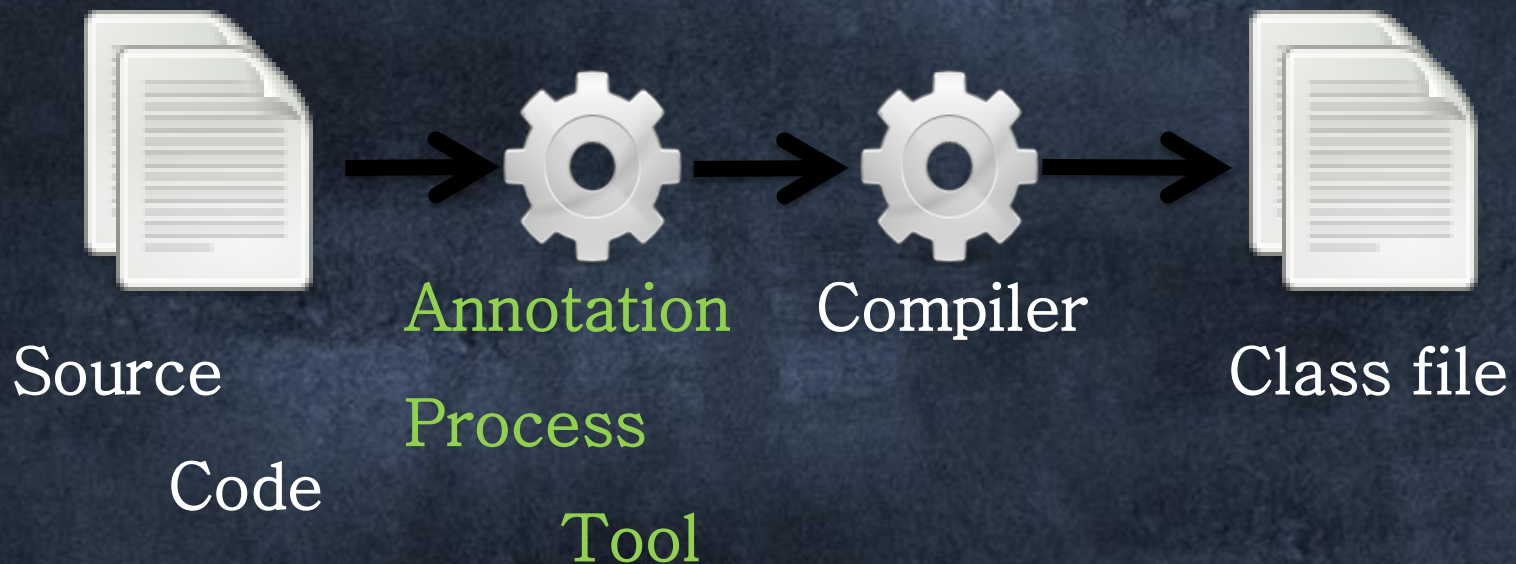
Reflection vs Code generation

APT(Annotation Process Tool)

JSR 269(JSR 269 Pluggable Annotation Processing API)

Annotation을 사용하여 소스 컴파일 전에

사용자가 원하는 작업을 할 수 있도록 하는 규격



excilys / androidann

androidannotations

<https://github.com/excilys/androidannotations>

Code

Network

Pull Requests 0

Issues 32

Wiki 56

Stats & Graphs

AndroidAnnotations - Simple and powerful Android code! — [Read more](#)<http://androidannotations.org>

ZIP



HTTP

Git Read-Only

<https://github.com/excilys/androidannotations.g>

Read-Only access

branch: develop

Files

Commits

Branches 5

Tags 7

Downloads

Latest commit to the **develop** branchMerge pull request [#90](#) from excilys/89_prevent_primitive_array

Pierre-Yves Ricau authored 1 day ago

commit e628613162

androidannotations /

name	age	message	history
AndroidAnnotations	3 days ago	Fixed issue #89 , prevent returning primitive and array type in GET an... [a-thomas]	
HelloWorldEclipse	December 29, 2011	Ignore IDEA-generated files [ealden]	
RoboGuiceExample	December 29, 2011	Ignore IDEA-generated files [ealden]	
androidannotations-dependencies	October 20, 2011	Added eclipse conf to enable debug in hotrepace in dev. Fixes issue 41 [pyricau]	
gradle	January 13, 2012	Upgraded to Gradle AndroidAnnotations Plugin 0.3.0 [ealden]	

RoboGuice 와 다르게

AndroidAnnotations

런타임이 아니라 컴파일 시,

모든 소스를 자동으로 생성함.

```
@EActivity(R.layout.translate)
```

```
public class TranslateActivity extends Activity {
```

```
    @ViewById EditText textInput;
```

```
    @AnimationRes Animation faceIn;
```

```
    @Click
```

```
    void doTranslate() {
```

```
        translateInBackground(textInput.getText().toString());
```

```
    }
```

```
    @Background
```

```
    void translateInBackground(String textToTranslate) {
```

```
        String translatedText = callGoogleTranslate(textToTranslate);
```

```
        showResult(translatedText);
```

```
    }
```


// DO NOT EDIT THIS FILE, IT HAS BEEN GENERATED USING AndroidAnnotations.


```
public final class TranslateActivity_ extends TranslateActivity
{
    @Override
    public void onCreate(Bundle savedInstanceState) {
        beforeCreate_(savedInstanceState);
        super.onCreate(savedInstanceState);
        setContentView(layout.main);
    }

    private void beforeCreate_(Bundle savedInstanceState) {
    }

    private void afterSetContentView_() {
        textView = ((TextView) findViewById(id.textView));
        .....
    }
}
```

AndroidAnnotations

ClassA.java를 Annotation Process Tool API를 통해 ClassA_.java 파일을 생성하여 처리한다.

1. AndroidManifest.xml 해당 액티비티명 + “_” 추가
2. Intent를 사용시에 작성한 액티비티명 + “_”를 추가하여야 함

AndroidAnnotations 기능

Cookbook – <https://github.com/excilys/androidannotations/wiki/Cookbook>

- Activities : @EActivity
- Application : @App
- View : @ViewById, @AfterViews
- Resource : @StringRes, @ColorRes, @AnimationRes
 - @BooleanRes
 - @ColorStateListRes
 - @DimensionRes
 - @DimensionPixelOffsetRes
 - @DimensionPixelSizeRes
 - @DrawableRes
 - @IntArrayRes
 - @IntegerRes
 - @LayoutRes
 - @MovieRes
 - @TextRes
 - @TextArrayRes
 - @StringArrayRes

AndroidAnnotations 기능

- Extras : @Extra
- SystemServices : @SystemService
- Injecting html : @HtmlRes, @FromHtml
- WorkingWithThreads : @Background, @UiThread, @UiThreadThreadDelayed
- HandlingEvents : @Click, @LongClick, @Touch, @ItemClick, @LongItemClick, @ItemSelect
- Handling options menu : @OptionsMenu, @OptionsItem
- REST API(With SpringAndroid) : @Rest, @RestService, @Get, @Put, @Post, @Delete, @Options, @Head, @Accept
- Trace Method Execution : @Trace

AndroidAnnotations 장단점

- 장점

1. Annotation Process Tool 이용하여 컴파일 시에 모든 코드가 생성되어 성능상 이점이 있다. (런타임 시 리플렉션을 사용하지 않음)
2. 다양한 커스텀 어노테이션이 제공됨

- 단점

1. 인텐트를 사용시 “_” 문자열 처리

Funf

[About](#)[Funf Journal](#)[Funf In a Box](#)[Developers](#)[Blog](#)[Contact](#)

Build a Funf app in 5 minutes.
No coding required with Funf In A Box!

[Get Started >](#)

About

- 안드로이드폰의 각종 센서를 쉽게 사용할 수 있는 오픈소스 프레임워크



Who is Funf for?

- Self – Tracking
 - Funf Journal 이라는 앱을 통해 원하는 센서 값만 받아 볼 수 있다. 개인적으로 자신의 데이터를 분석하는데 사용
- Study Managers/Researchers
 - 센서를 통해 얻는 여러 정보들을 모아서 분석하고 연구
- API-Level Developers
 - Funf의 내부단을 사용하지 않고 3rd-party 개발자 API만 사용하여 개발 할 수 있도록 해준다
- Core-Level Developers
 - Funf에 없는 새로운 센서를 추가하거나 기존에 있는 센서 측정의 performance를 높인다

Funf probe List

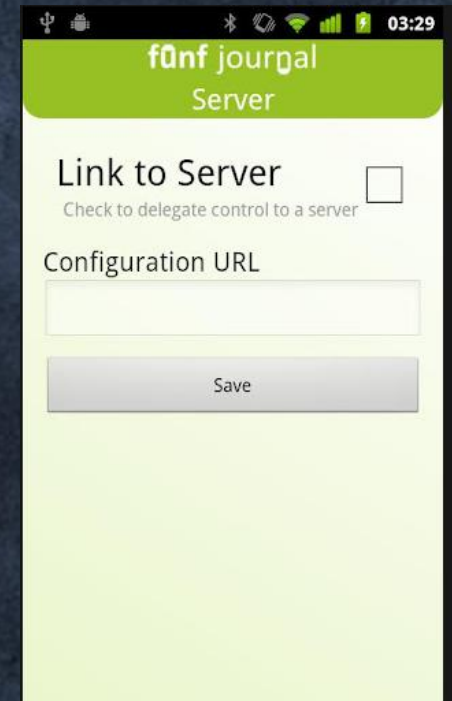
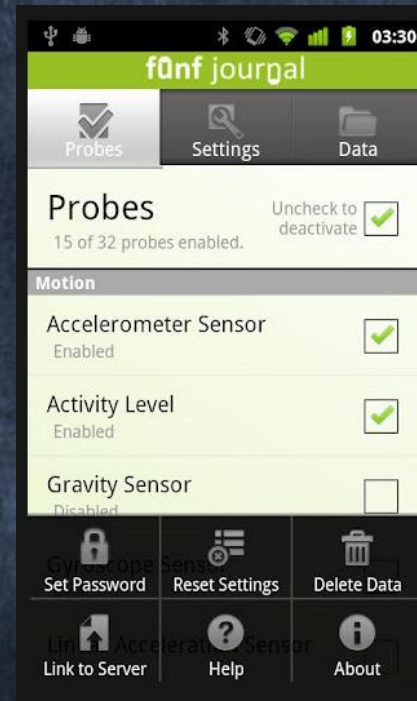
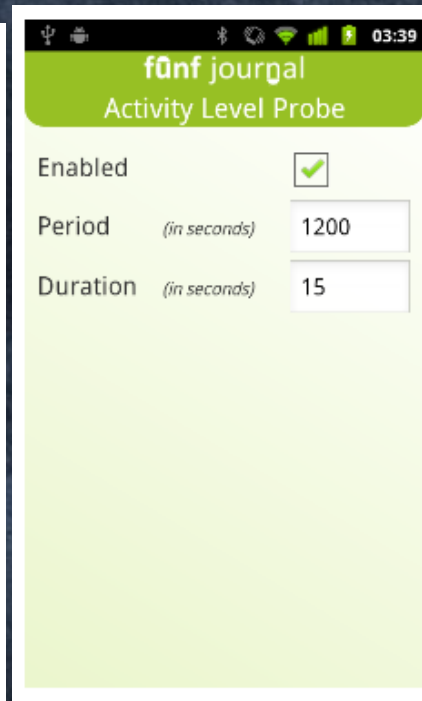
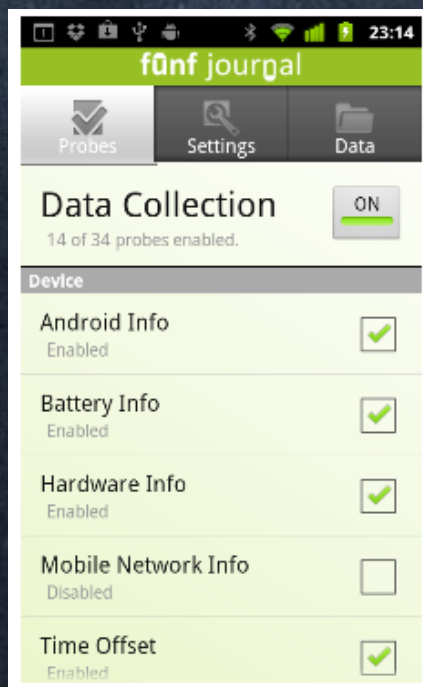
- Physical sensor(Hardware), Logical sensor(Software)
→ Probe

- Positioning
 - Location
 - Bluetooth
 - Wifi
 - Cell
- Social
 - Contact
 - Call Log
 - SMS
- Motion
 - Accelerometer Sensor
 - Gravity Sensor
 - Linear Acceleration Sensor
 - Gyroscope Sensor
 - Orientation Sensor
 - Rotation Vector Sensor
 - Activity

- Environment
 - Light Sensor
 - Proximity Sensor
 - Magnetic Field Sensor
 - Pressure Sensor
 - Temperature Sensor
- Device
 - Android Info
 - Battery
 - Hardware Info
 - Time Offset
 - Telephony
- Device Interaction
 - Running Applications
 - Applications
 - Screen
 - Browser Bookmarks
 - Browser Searches
 - Videos
 - Audio Files
 - Images

Funf journal(Self-Tracker)

- Funf framework를 이용해 만든 Application
- 간단하게 원하는 센서들의 데이터 수집 가능
- 수집주기, 데이터 전송 위치 설정 가능



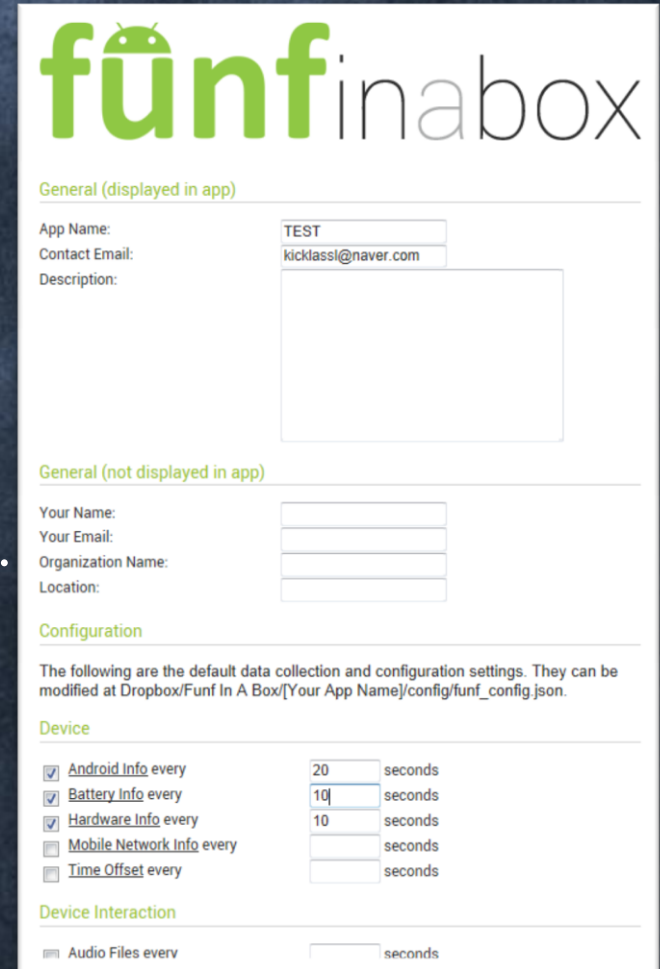
Funf journal

- 수집된 데이터는 *.db 파일이며, 암호화 되어있다
 - 디코딩 프로그램제공(http://funf.media.mit.edu/downloads/funf_analyze_win.zip)
 - JSON 형태의 결과 데이터

```
{  
  "icon-small": 17302188,  
  "present": true,  
  "scale": 100,  
  "level": 53,  
  "technology": "Li-ion",  
  "status": 2,  
  "voltage": 3981,  
  "PROBE": "edu.mit.media.funf.probe.builtin.BatteryProbe",  
  "plugged": 1,  
  "TIMESTAMP": 1333358639,  
  "health": 2,  
  "temperature": 310  
}
```

Funf in a box

- Researcher 용으로 여러 테스터들에게서 원하는 센서 데이터들을 수집하는데 사용
- 간단하게 데이터 수집용 앱을 생성
- APP 이름, 데이터를 받을 e-mail, 수집할 센서정보를 세팅 한 후 생성하면 .apk 파일을 얻을 수 있다.



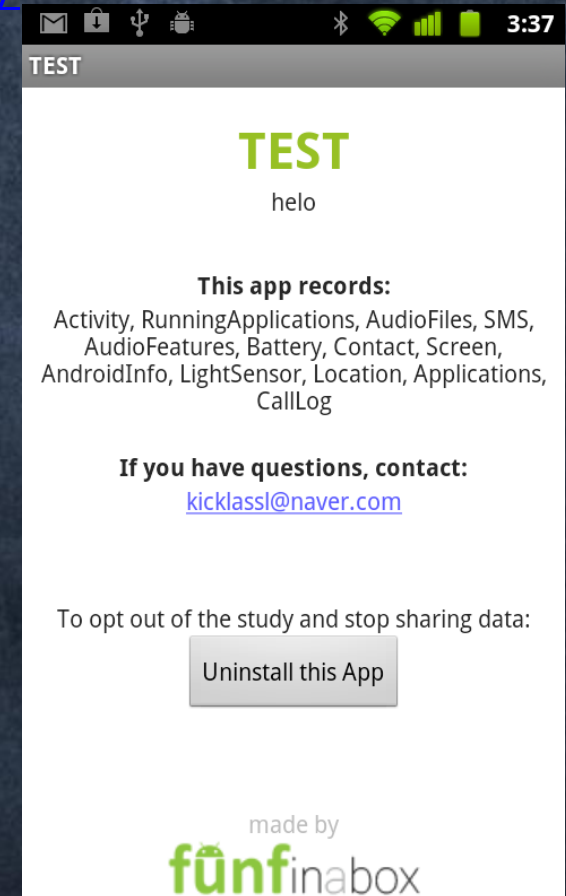
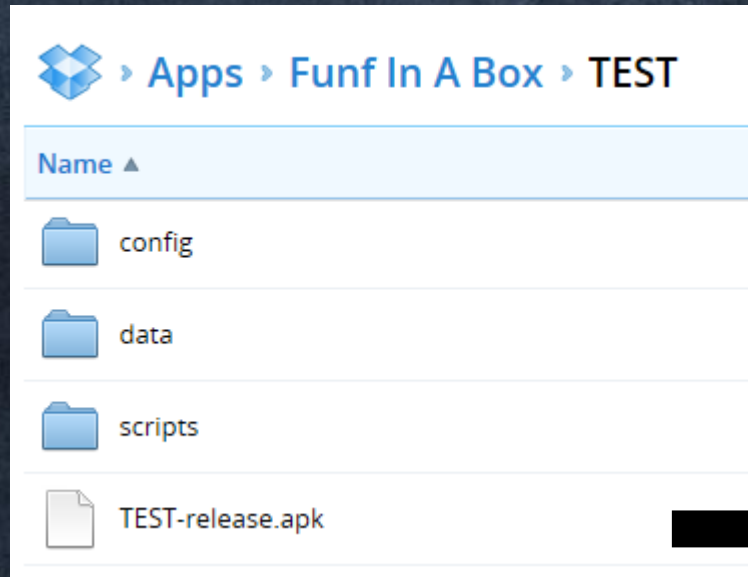
The screenshot displays the 'Funf in a box' web application interface. At the top, the logo 'funf in a box' is shown with a green Android robot head above the 'u' in 'funf'. Below the logo, the page is divided into several sections:

- General (displayed in app)**: This section contains three input fields: 'App Name' (with the value 'TEST'), 'Contact Email' (with the value 'kicklassi@naver.com'), and 'Description' (a large empty text area).
- General (not displayed in app)**: This section contains four input fields: 'Your Name', 'Your Email', 'Organization Name', and 'Location'.
- Configuration**: This section contains a paragraph of text: 'The following are the default data collection and configuration settings. They can be modified at Dropbox/Funf In A Box/[Your App Name]/config/funf_config.json.'
- Device**: This section contains five rows of configuration options, each with a checkbox, a label, and a text input field for the interval in seconds:
 - ☒ **Android Info** every 20 seconds
 - ☒ **Battery Info** every 10 seconds
 - ☒ **Hardware Info** every 10 seconds
 - ☐ **Mobile Network Info** every [] seconds
 - ☐ **Time Offset** every [] seconds
- Device Interaction**: This section contains one row of configuration options:
 - ☐ **Audio Files** every [] seconds

Funf in a box

- .apk 파일은 Dropbox를 통해서 받을 수 있다

–  <https://www.dropbox.com/>



Key feature

- 수집된 데이터는 JSON 형태
- 수집데이터를 원격지에 주기적으로 보낼 수 있다
- 약 30여개의 probe 제공
- 암호화를 통해 민감한 정보를 숨김(폰번호, 이름 등)
- 로컬 저장하는 DB 파일은 암호화되어 저장됨
- 15개월의 테스트를 통해 프레임워크를 검증
- 배터리 소모량 최적화

Tutorial

- Google code에 source 및 각종설명
 - <http://code.google.com/p/funf-open-sensing-framework/wiki/GettingStarted?tm=6>(wiki)
- Git 을 통해 소스 배포
 - <https://code.google.com/p/funf-open-sensing-framework.samples>(샘플)
 - <https://code.google.com/p/funf-open-sensing-framework> (funf framework)
- Jar file로도 제공
 - <http://code.google.com/p/funf-open-sensing-framework/downloads/list>

Tutorial

- Probe 정보

- <http://code.google.com/p/funf-open-sensing-framework/wiki/BuiltinProbes>

- Probe parameter

- Period
 - Duration
 - Start
 - End

Battery

Information about the type and current state of the battery in the device.

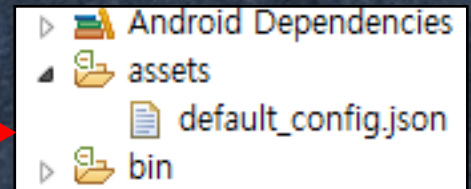
- **Name:** edu.mit.media.funf.probe.builtin.BatteryProbe
- **Required Permissions:** android.permission.BATTERY_STATS
- **Required Features:** none
- **Parameters:**
 - PERIOD (default 300s)
 - START
 - END
- **Example Data:**

```
{
  "TIMESTAMP": 1316035006,
  "icon-small": 17302188,
  "present": true,
  "scale": 100,
  "level": 94,
  "technology": "Li-ion",
  "status": 2,
  "voltage": 4176,
  "plugged": 2,
  "health": 2,
  "temperature": 280
}
```


Tutorial

- Probe 사용법

- Asset/default_config.json 에 설정하기



```
{  "name": "TestScanner",
    "version": 1,
    "dataArchivePeriod": 30,
    "dataRequests": {
        "edu.mit.media.funf.probe.builtin.ScreenProbe": [
            {}
        ],
        "edu.mit.media.funf.probe.builtin.RunningApplica
            { "PERIOD": 20 }
        ],
        "edu.mit.media.funf.probe.builtin.LocationProbe":
            { "PERIOD": 30, "DURATION": 10 }
    }
}
```

Tutorial(Example)

- Asset/default_config.json 에 설정 후
- 사용할 서비스와 퍼미션 등록

```
<!-- Probe Services -->  
<service android:name="edu.mit.media.funf.probe.builtin.LocationProbe"></service>  
<service android:name="edu.mit.media.funf.probe.builtin.WifiProbe"></service>
```

```
<!-- Launching -->  
<uses-permission android:name="android.permission.BATTERY_STATS" />  
  
<!-- All probes -->  
<uses-permission android:name="android.permission.WAKE_LOCK"/>  
  
<!-- Location probe -->  
    <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION"/>  
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>  
  
<!-- Wifi probe -->  
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE"/>  
<uses-permission android:name="android.permission.CHANGE_WIFI_STATE"/>
```


Tutorial(Example)

- 서비스 실행

- MainPipeLine.class 에서 config 정보 읽어서 세팅함

```
Intent archiveIntent = new Intent(context, MainPipeline.class);  
String action = MainPipeline.ACTION_ENABLE;  
archiveIntent.setAction(action);  
startService (archiveIntent);
```

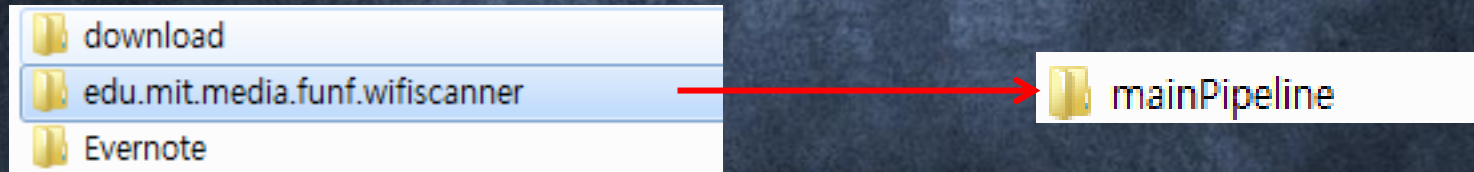
- MainPipeLine 코드 중 getMainConfig()

- onHandleIntent()에서 getMainConfig() 실행

```
public static FunfConfig getMainConfig(Context context) {  
    FunfConfig config = getConfig(context, MAIN_CONFIG);  
    if (config.getName() == null) {  
        String jsonString = getStringFromAsset(context, "default_config.json");
```

Tutorial(Example)

- Sdcard/edu.mit.media.funf.wifiscanner 에 저장됨



- Funf_analyze 프로그램을 같은 폴더에 넣고 실행
 - Default 비밀번호 : changeme


```
G:\edu.mit.media.funf.wifiscanner\funf_analyze.exe
Unzipping ['G:\edu.mit.media.funf.wifiscanner\202694b5-4480-42c6-a20b-b14403e676c9_1334136023_mainPipeline.zip']
Funf data files are encrypted. Please enter the password you entered on the device to decrypt them.
Enter encryption password:
```


Tutorial(Example)

- Merged_data.db 생성됨

id	device	probe	timestamp	value
434bcde1-e5c0-4	202694b5-4480-42	edu.mit.media.fu	1334135984	{...
434bcde1-e5c0-4	202694b5-4480-42	edu.mit.media.fu	1334136017	{...
434bcde1-e5c0-4	202694b5-4480-42	edu.mit.media.fu	1334136017	{...

- Id : unique id for the data entry
- Device : UUID for the device
- Timestamp : long, UTC in seconds since epoch
- Value : result data in json format



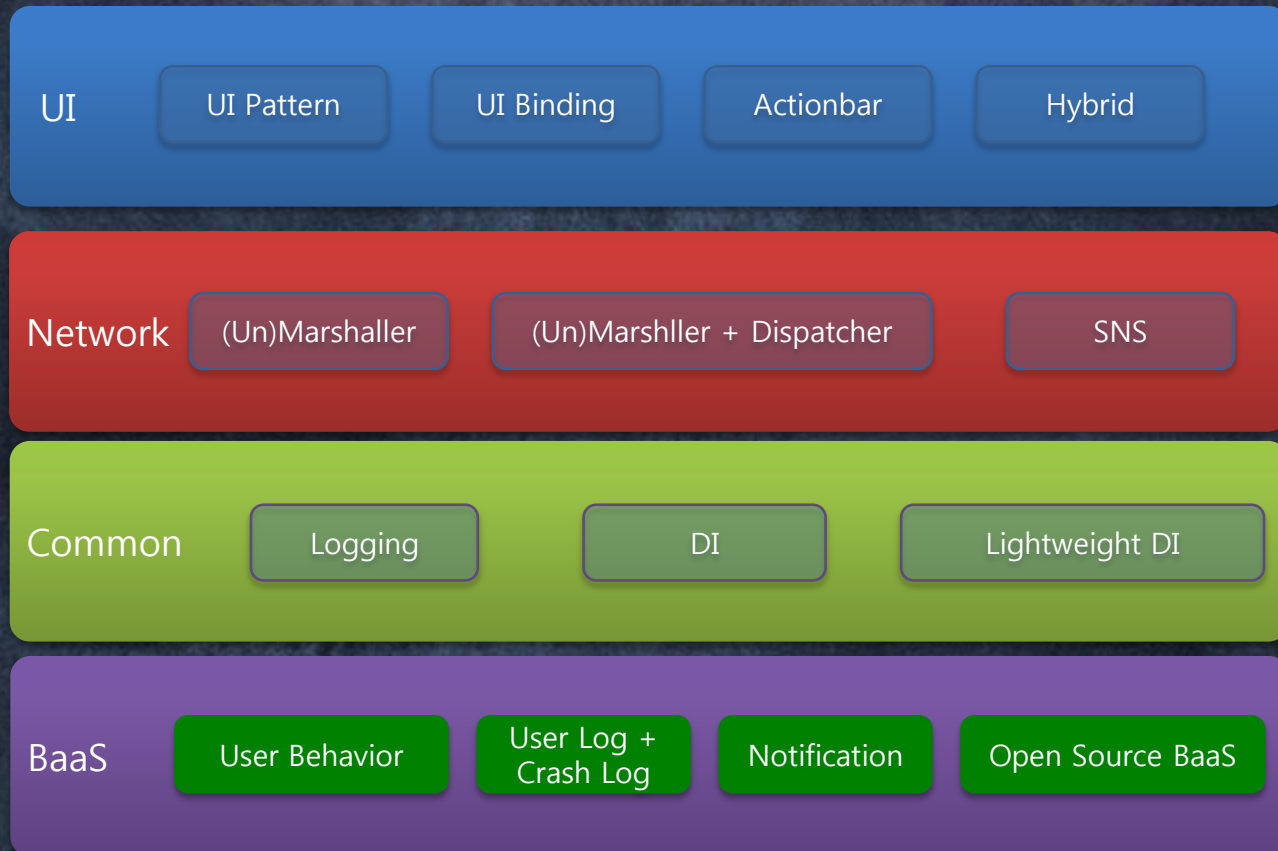
```
{
  "icon-small": 17302188,
  "present": true,
  "scale": 100,
  "level": 53,
  "technology": "Li-ion",
  "status": 2,
  "voltage": 3981,
  "PROBE": "edu.mit.media.funf.probe.builtin.BatteryProbe",
  "plugged": 1,
  "TIMESTAMP": 1333358639,
  "health": 2,
  "temperature": 310
}
```

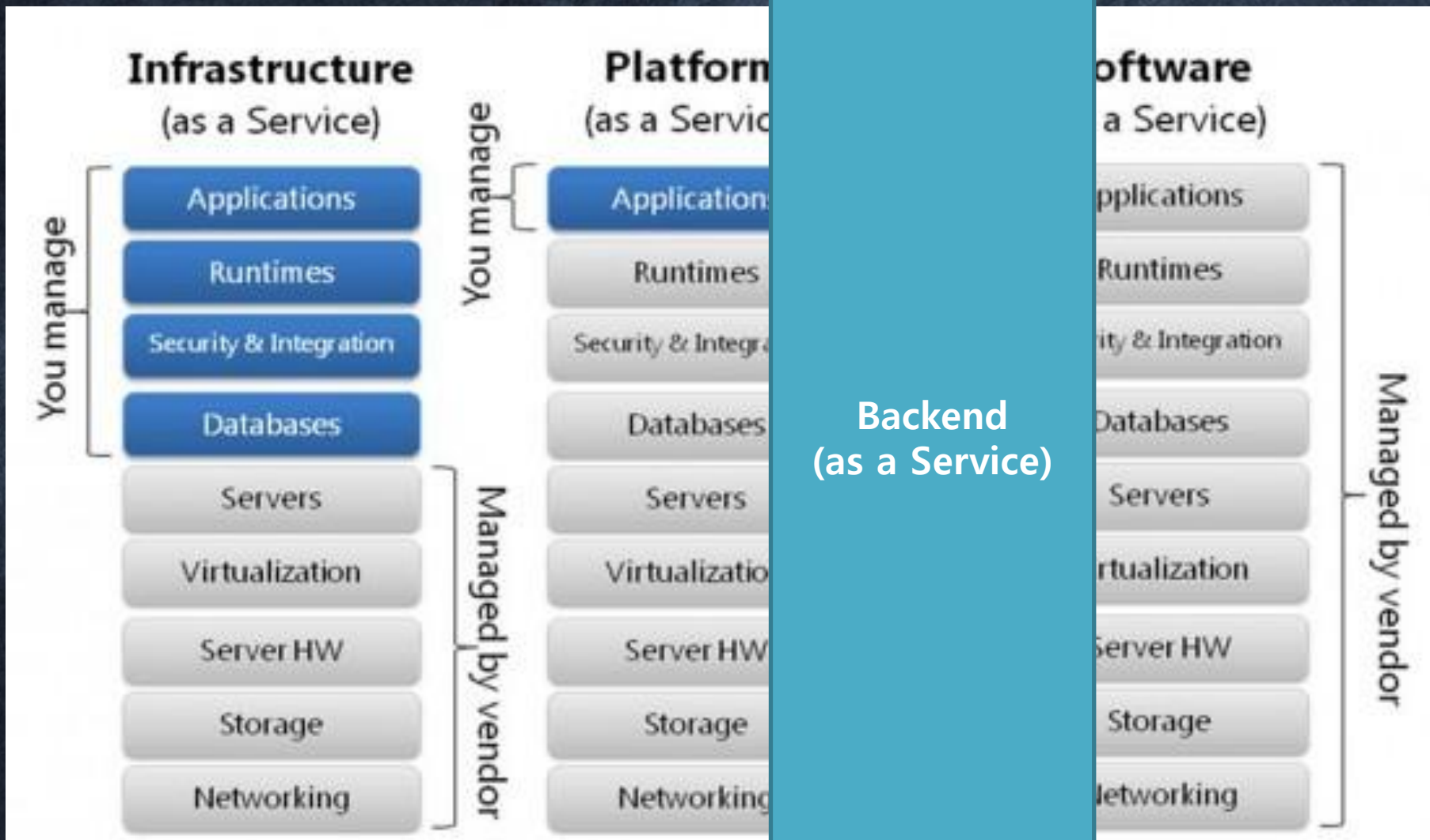
Part 4. BaaS

Backend As A Service



Android Application Block





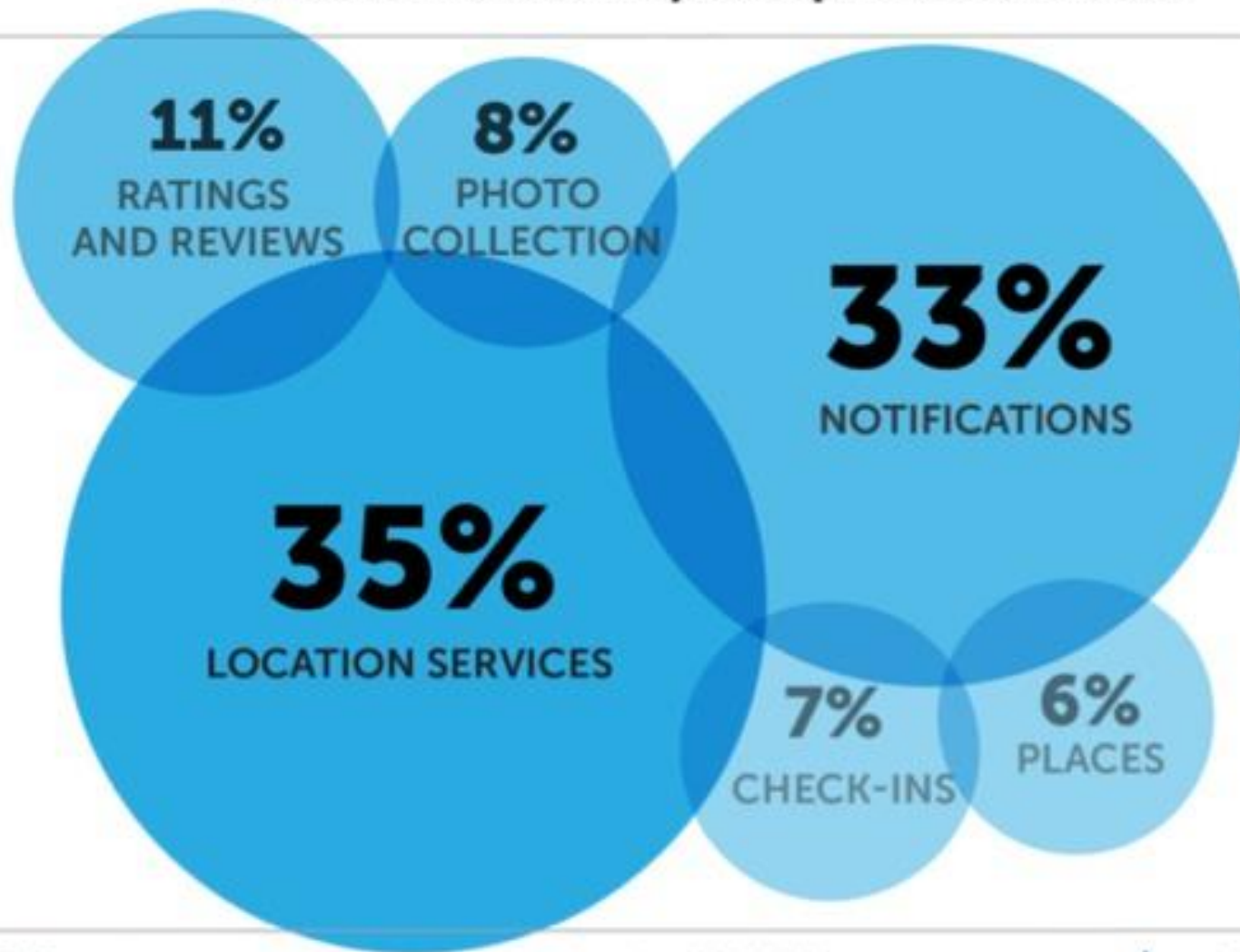
BaaS(Backend As A Service)

서버 개발을 모르는 클라이언트 개발자
도

쉽게 서버가 필요한 앱을 만들 수 있게
도와주는 서비스

What are the top new or existing cloud-based services you plan to connect or extend?

Cloud Services Developers Expect to Use in 2012



N=585 responses

Source: Appcelerator / IDC - 01/2012



Backend features

OAuth

Custom Data

Social

Counters

Users

Events

Collections

Activities

Groups

Feeds

Entities

Queries

Roles

File support

Push notify

RESTful

Permissions

Geolocation

Relationships

Errors

Parse

Products

Customers

Pricing

Docs

About

Blog

Sign Up

Log In

We're
hiring

9장. Parse

The complete



<https://www.parse.com/>

Parse lets you focus on creating unique & engaging apps. We take care of the rest.

Compatible
iOS   HTML5

Power your app using Parse





Try it for free

No credit card required

CUSTOMERS



월 100만건 호출까지는 무료...

	 Basic \$0 per month	 Pro \$199 per month <div>30-day free trial</div>	 Enterprise
API Requests  per month	1,000,000 7¢ per 1,000 over	15,000,000 5¢ per 1,000 over	<p>Need more than Pro offers? Contact us for a custom plan and pay rates much lower than the overages!</p> <p>(877) 747-8568 enterprise@parse.com</p>
Pushes  per month	1,000,000 7¢ per 1,000 over	5,000,000 5¢ per 1,000 over	
File Storage  per month	1GB 20¢ per 1GB over	10GB 15¢ per 1GB over	
Roles for Access Control 	1	Unlimited	
App Collaborators 		2 \$15/extra	

데이터 저장하기

Key-value 된 값을 저장

```
ParseObject gameScore = new  
ParseObject("GameScore");  
gameScore.put("score", 1337);  
gameScore.put("playerName", "Sean Plott");  
gameScore.put("cheatMode", false);  
gameScore.saveInBackground();
```


데이터 저장하기

Data Browser 통해서 확인가능

The screenshot shows the Parse Data Browser interface. At the top, the Parse logo is visible along with links to Dashboard, Quickstart, and Documentation. Below this, there's a navigation bar with a hamburger menu, a dropdown menu set to 'softwaregeeks', and tabs for Overview, Data Browser (which is active), and Cloud Code. The main content area is titled 'Classes' and includes buttons for '+ Row', '- Row', '+ Col', '- Col', and a 'More' dropdown. On the left, a list of classes is shown with their respective counts: Installation (1), Role (0), User (4), TestObject (6), TestObject222 (1), Todo (3), and tweets (0). The 'TestObject' class is selected, and its data is displayed in a table. The table has columns for 'objectId', 'foo', and 'createdAt'. The data rows show various object IDs and their corresponding 'foo' values and creation dates.

objectId	foo	createdAt
zqu4Y1RSzC	(empty)	Sat, 17 Nov 2012
G3I7VMuv4v	bar	Thu, 06 Sep 2012
dyD1G60ajQ	bar	Thu, 06 Sep 2012
xMpH5wrKAg	bar	Thu, 06 Sep 2012
mEzhvrDKNs	bar	Thu, 06 Sep 2012
It2XrLPLQE	bar	Thu, 06 Sep 2012

푸시설정

```
PushService.subscribe(this, "", YourActivity.class);
```

```
<service android:name="com.parse.PushService" />
<receiver android:name="com.parse.ParseBroadcastReceiver">
    <intent-filter>
        <action android:name="android.intent.action.BOOT_COMPLETED" />
        <action android:name="android.intent.action.USER_PRESENT" />
    </intent-filter>
</receiver>
```

```
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission
android:name="android.permission.RECEIVE_BOOT_COMPLETED" />
<uses-permission android:name="android.permission.VIBRATE" />
<uses-permission
android:name="android.permission.ACCESS_NETWORK_STATE" />
```


푸시보내기

[Back](#)

Send a Push

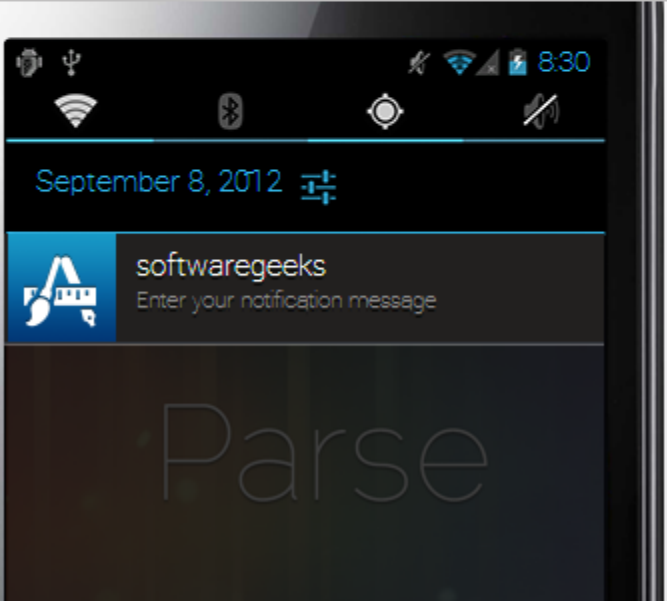
Send to: Everyone 1 recipient

Device Types: ☐ iOS clients ☒ Android clients

Compose message

Enter your notification message

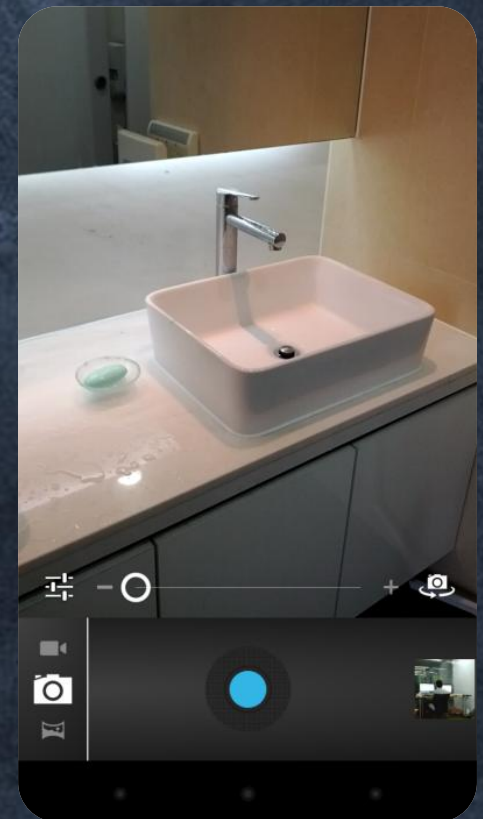
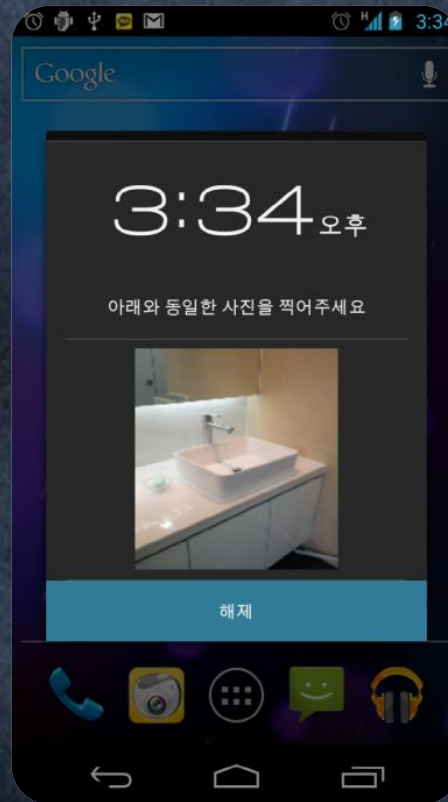
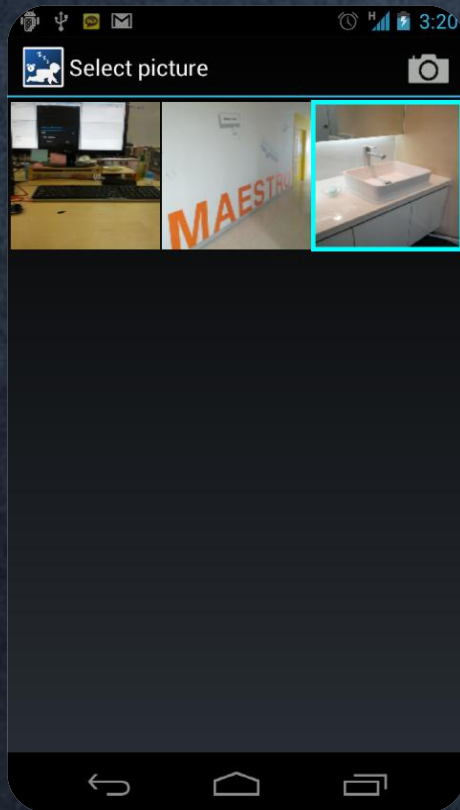
[Message](#) ☐ JSON 180 character(s) left

A mockup of an Android phone screen. At the top, the status bar shows various icons and the time 8:30. Below that, a date separator shows 'September 8, 2012'. A notification from 'softwaregeeks' is displayed with the text 'Enter your notification message'. The background of the phone screen shows the 'Parse' logo.

10장. 사용자 행동 분석



Sleep If you can.



Issue..

무조건 깬 수밖에 없는 알람 앱 화제



片，可
要返回
鬧鐘是不

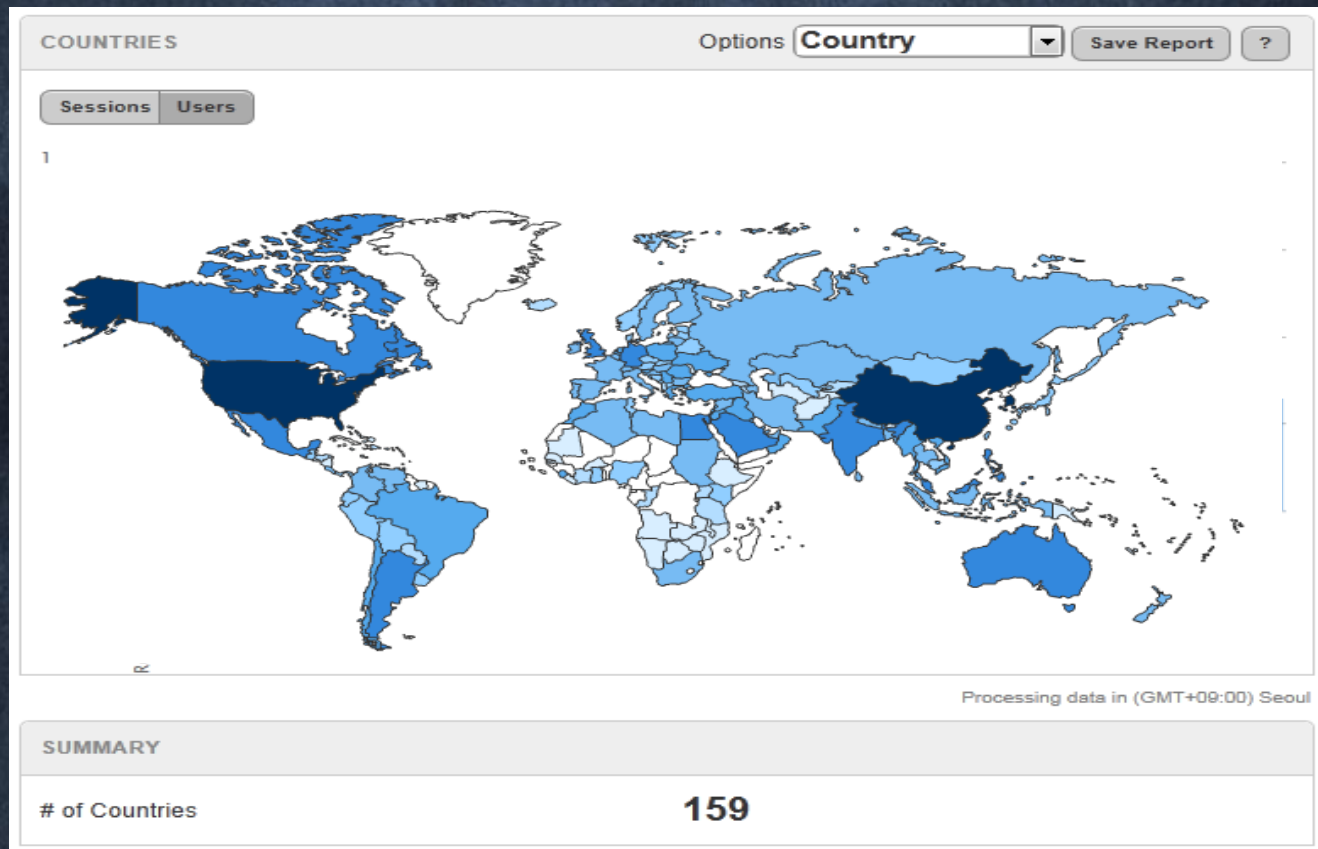
무조건 깬 수밖에 없는 알람 앱

3:34

클라우드 러너
MobAppBox

Cash King (Make
Money)
Byte Ventures

월드 와이드 앱



모바일 어플리케이션 분석 플랫폼

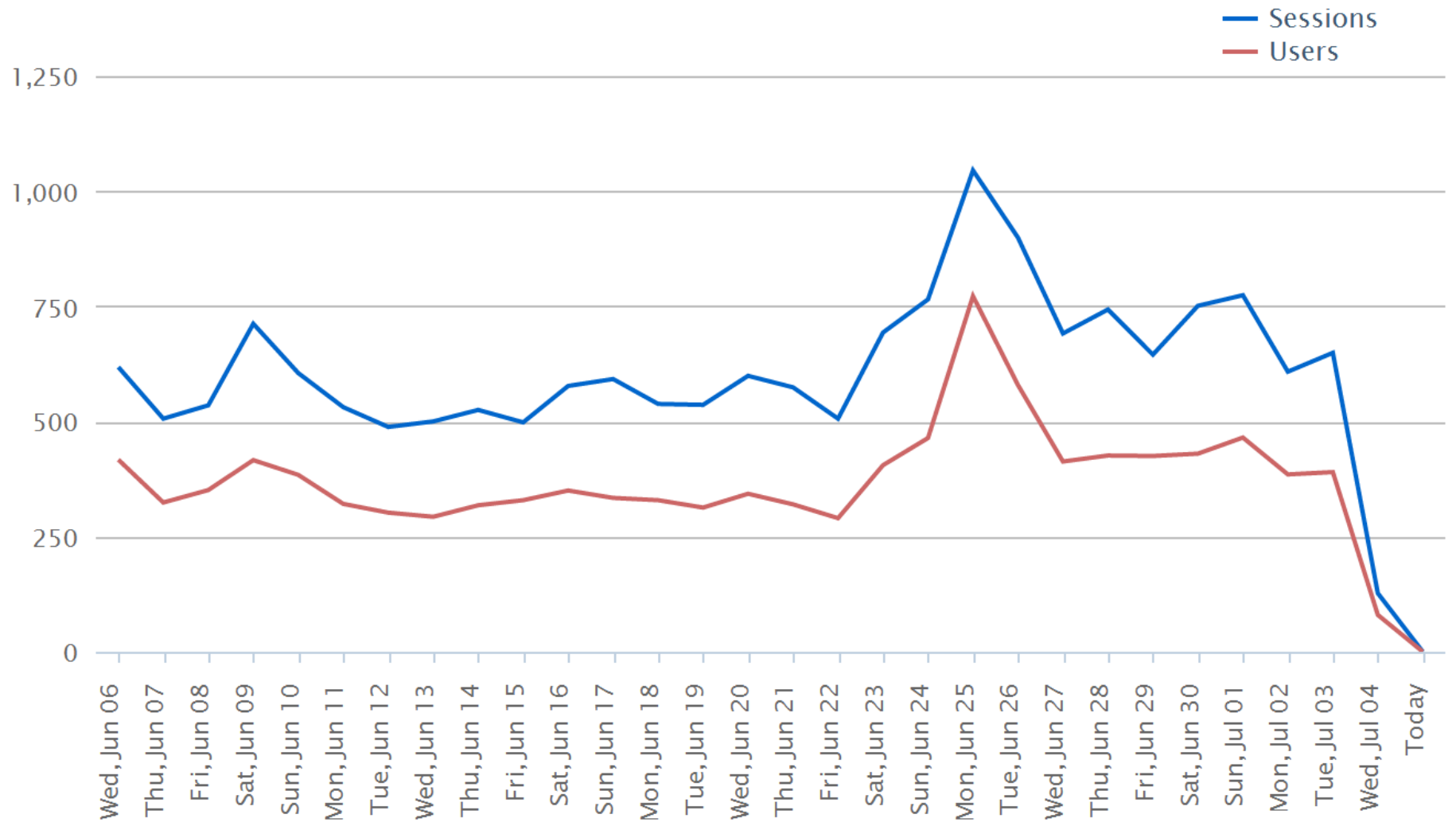
Localytics



메인 DASHBOARD

DASHBOARD

?



Daypart

NEW VS RETURNING BY HOUR

Scale to 100%: ☐

Options:

New vs Returning 

Sessions

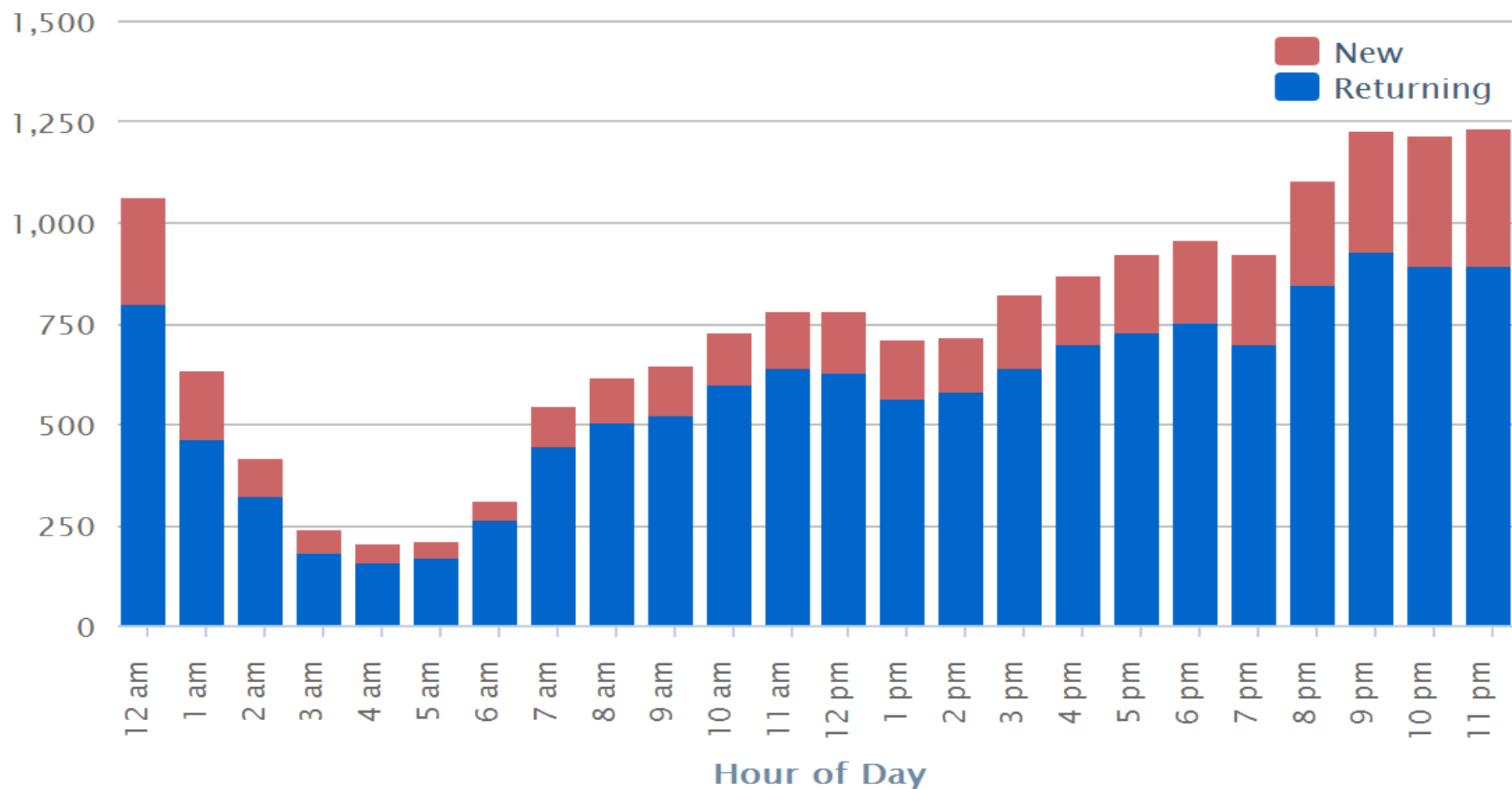
Users

Hour

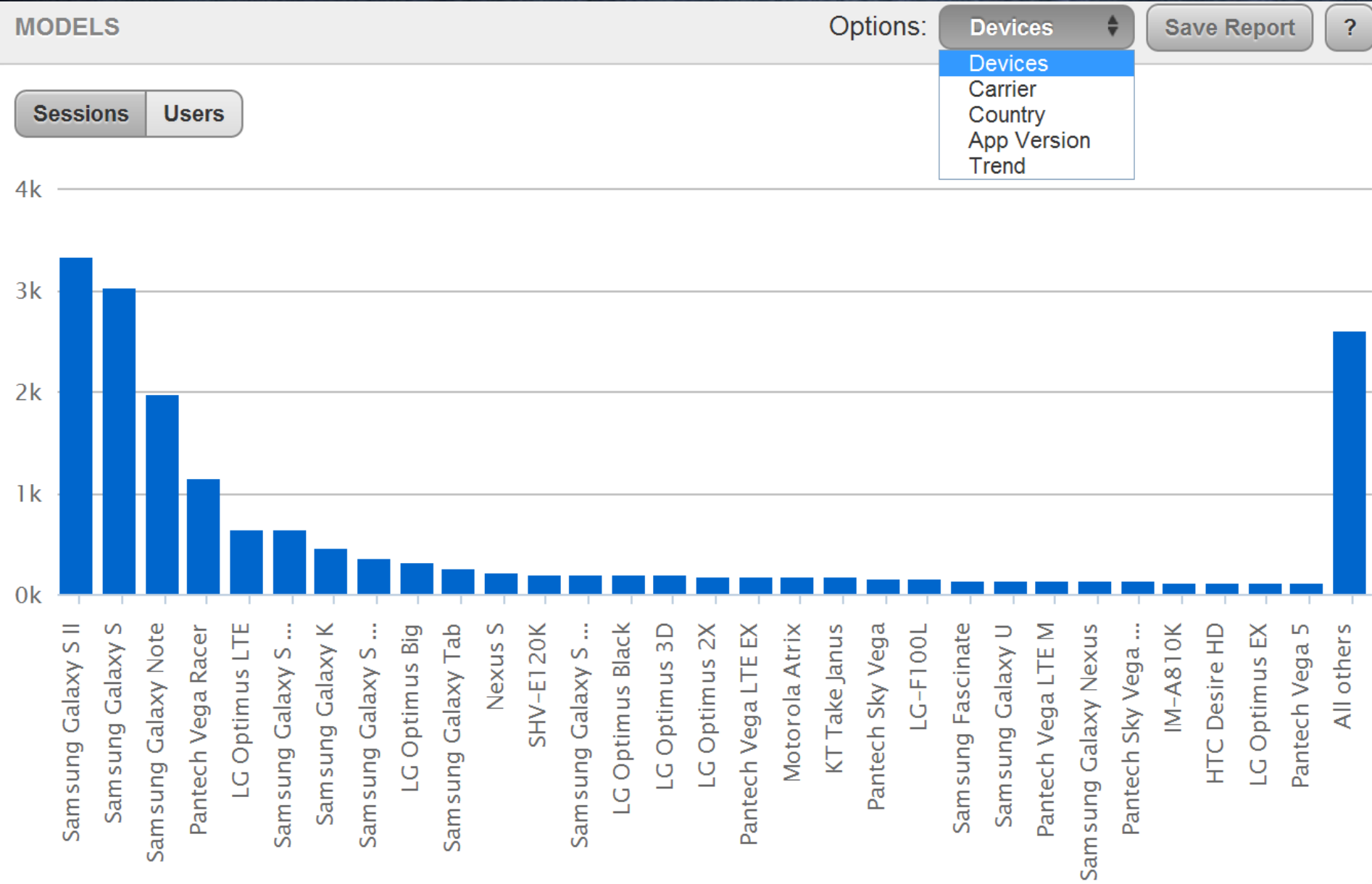
New vs Returning

Countries

Application Version



Device (국내는...)



Device (월드 와이드 앱은..)

DEVICES

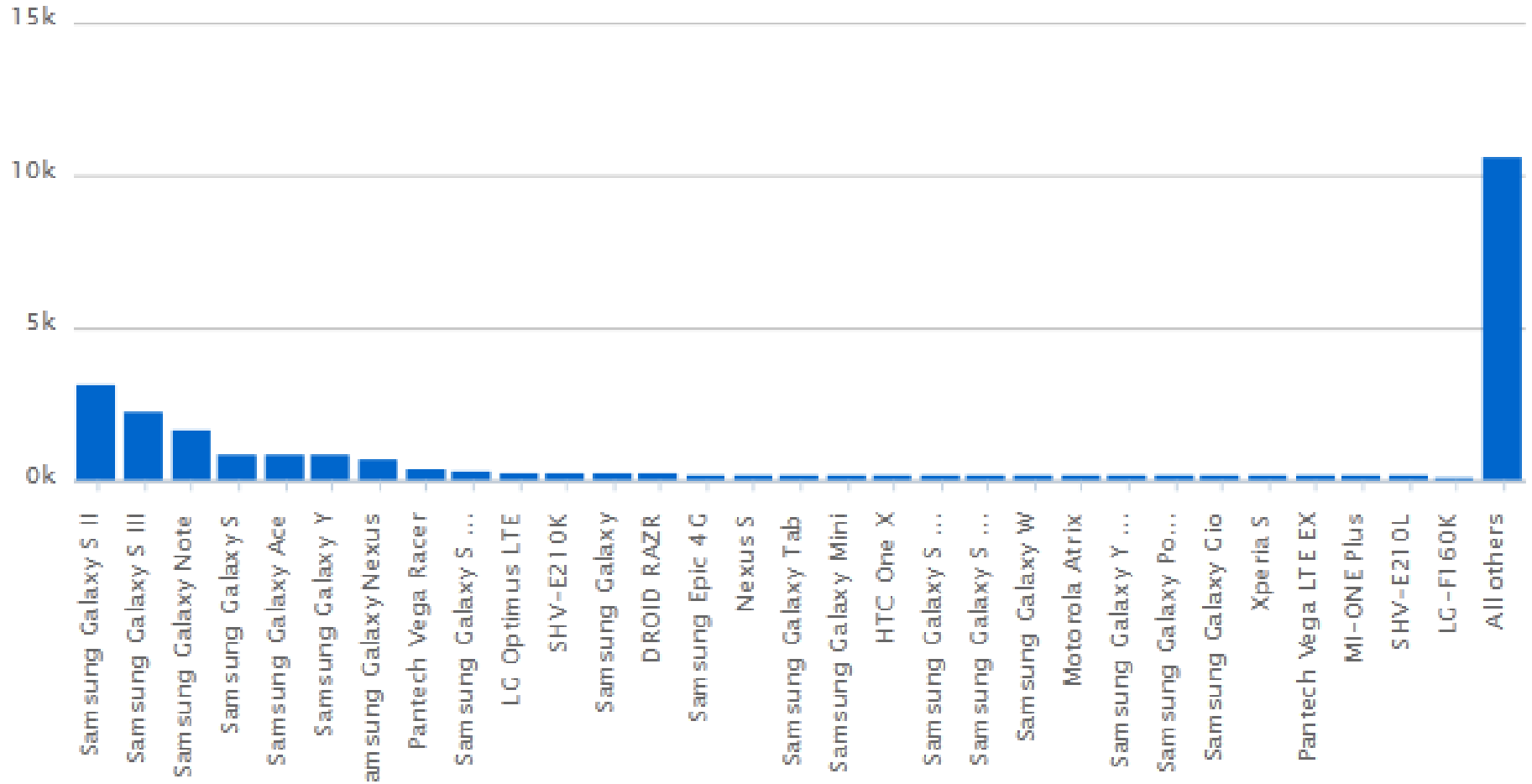
Options **Devices**

Save Report

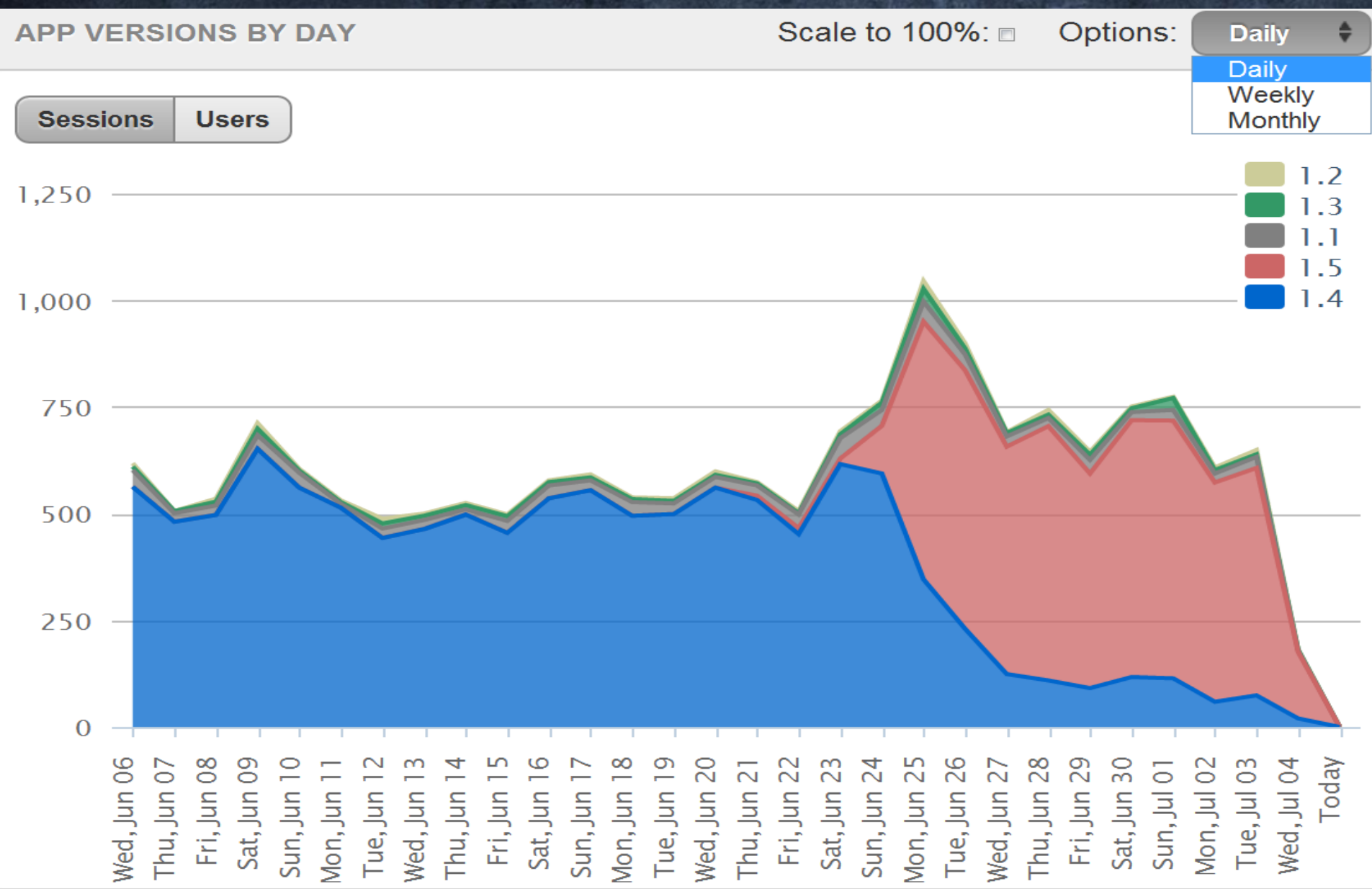
?

Sessions

Users



App Version



TagEvent (사용자가 많이쓰는 기능)

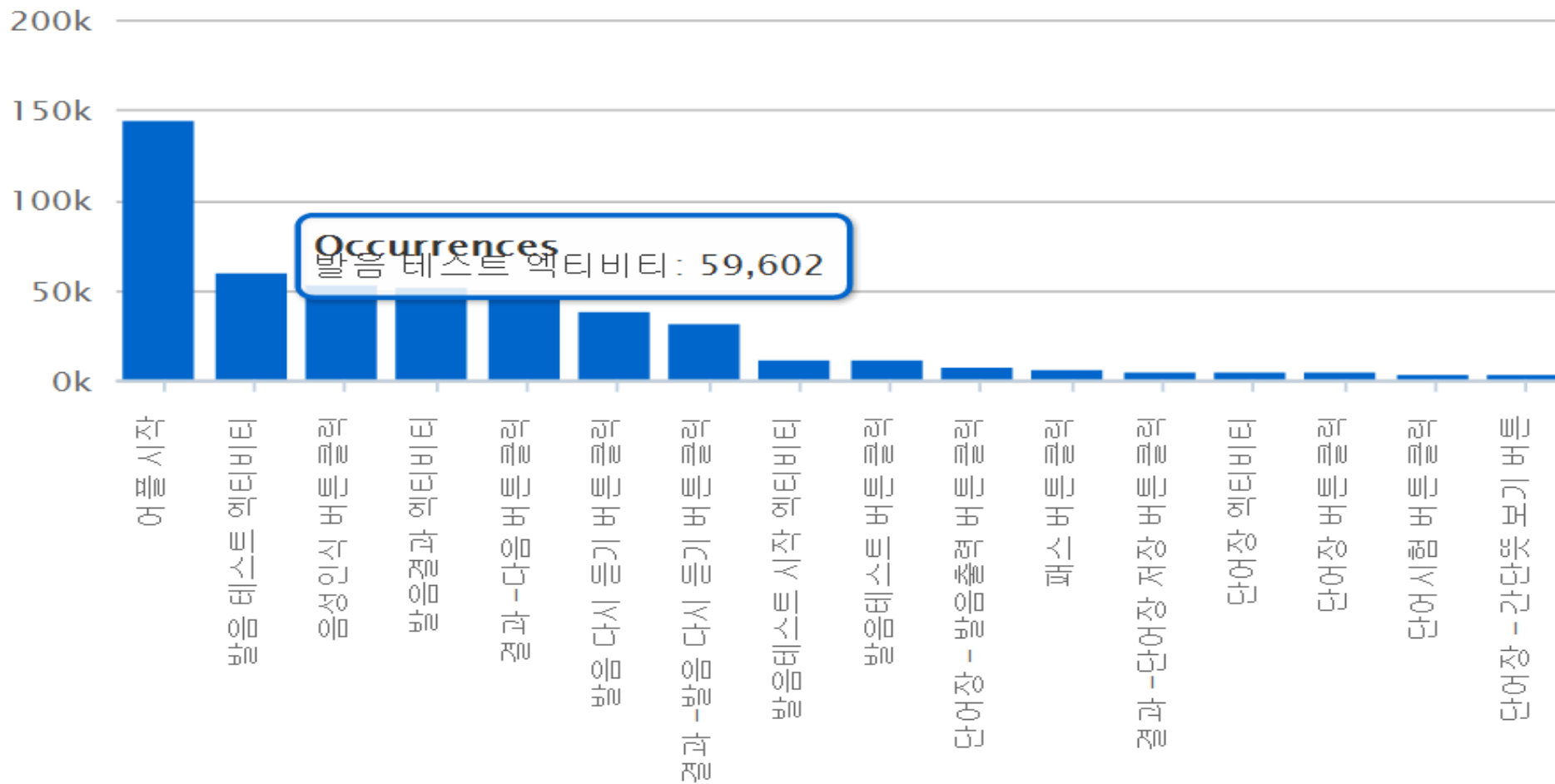


Options:

Event counts



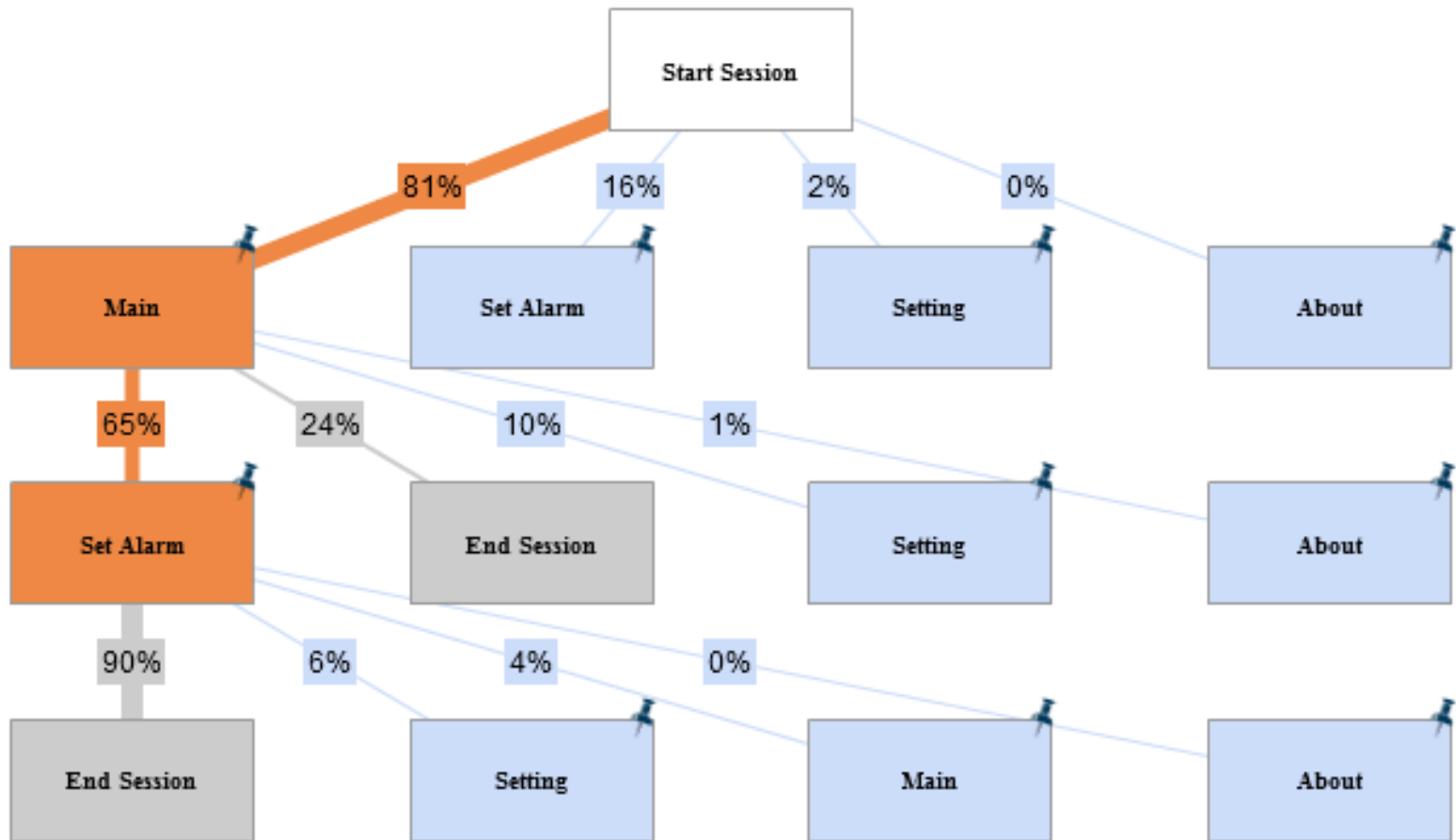
Save Report



사용자가 어떤 화면을 많이 사용하는지 (유료기능)

SCREEN FLOW OVERVIEW

?



이외에도..

- **New vs Returning**
- **Carrier**
- **Country**
- **OS Version**

- <http://www.localytics.com/>에서 **Sign Up**

- **Administration** 탭 에서 어플리케이션 등록

PRONOUNCIATIONNOTE

EditPermissionsDelete

Category: Education

Key: fc2743b1b6697feb822c69c-08f3f1b2-fe47-11e0-8ea1-007bc6310ec9


Created: 2011-10-24 13:49:46 UTC


Feature level: Community


Time zone: (GMT+00:00) UTC


Upgrade


For a 10 minute integration guide, click the appropriate link below.


ANDROID


HTML

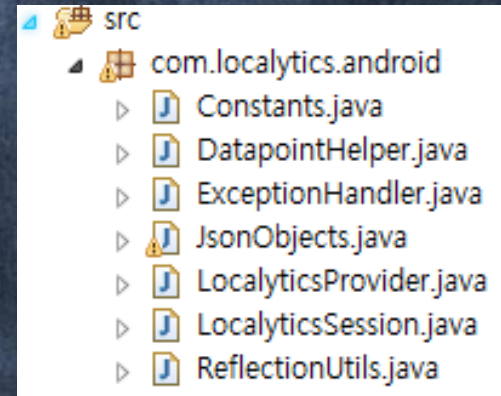

BlackBerry


HTML


HTML5

- <http://bit.ly/KnZzpx> 에서 소스코드 다운

- Android 프로젝트
src/폴더에 소스코드 추가



- AndroidManifest.xml에 퍼미션 추가

<uses-permission

android:name="android.permission.INTERNET">

- Main Activity에 Import

```
import com.localytics.android.*;
```

- 클래스 내부에 session object 추가

```
private LocalyticsSession localyticsSession;
```

- OnCreate 함수에 아래 소스 추가

```
public void onCreate(Bundle savedInstanceState)
{
    :
    this.localyticsSession = new LocalyticsSession(this.getApplicationContext(),
        "APP KEY FROM STEP 2");
    this.localyticsSession.open(); // open the session
    this.localyticsSession.upload(); // upload any data
    :
}
```

- onResume 메소드 Override

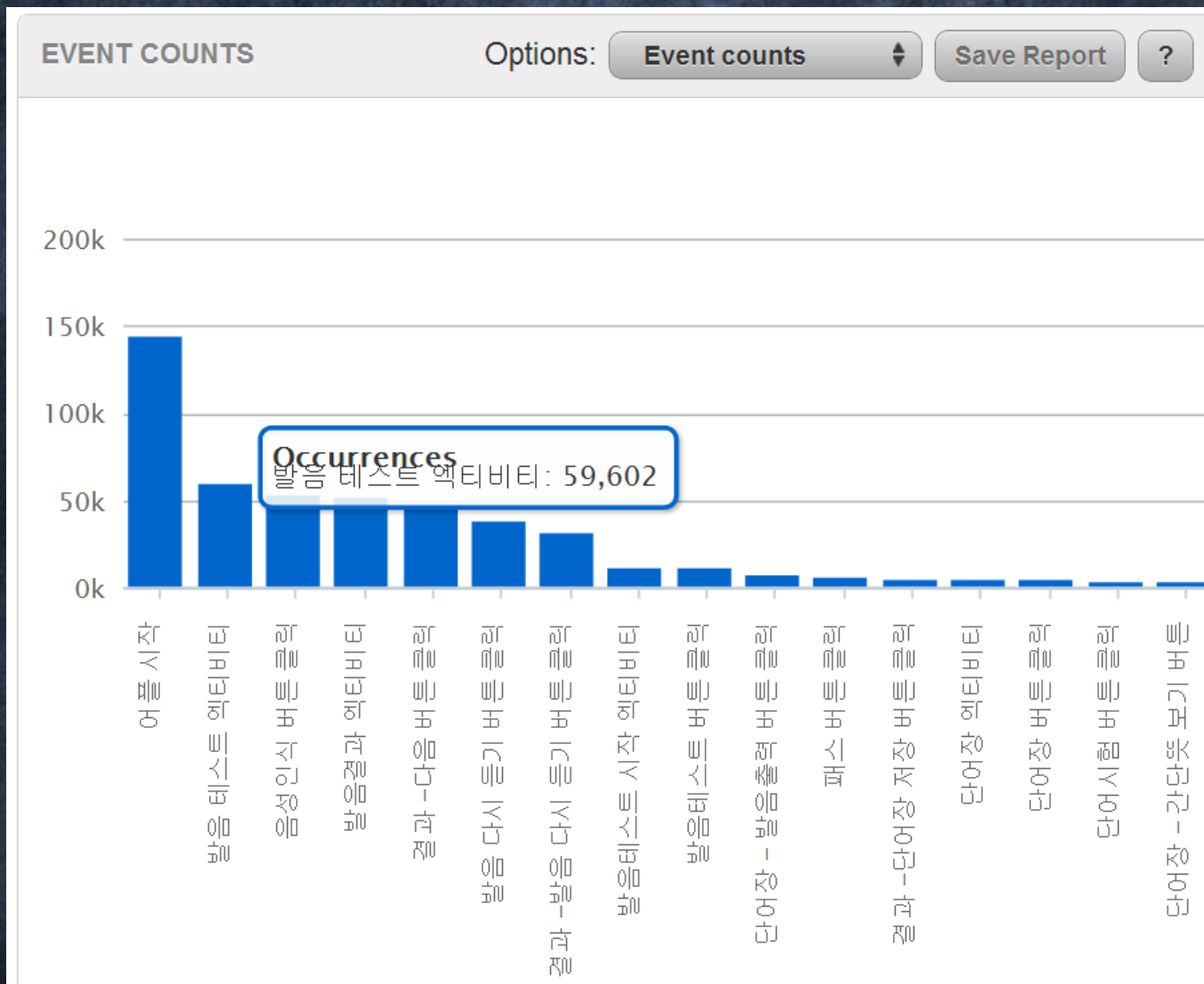
```
public void onResume()  
{  
    super.onResume();  
    this.localyticsSession.open();  
}
```

- onPause 메소드 Override

```
public void onPause()  
{  
    this.localyticsSession.close();  
    this.localyticsSession.upload();  
    super.onPause();  
}
```


- TagEvent남기기

localyticsSession.tagEvent("TagEvent남김");



다양한 BaaS 제공되고 있기 때문에
모바일 개발자 입장에서 서비스를 택해
서
개발을 쉽게 할 수 있음

Q&A

