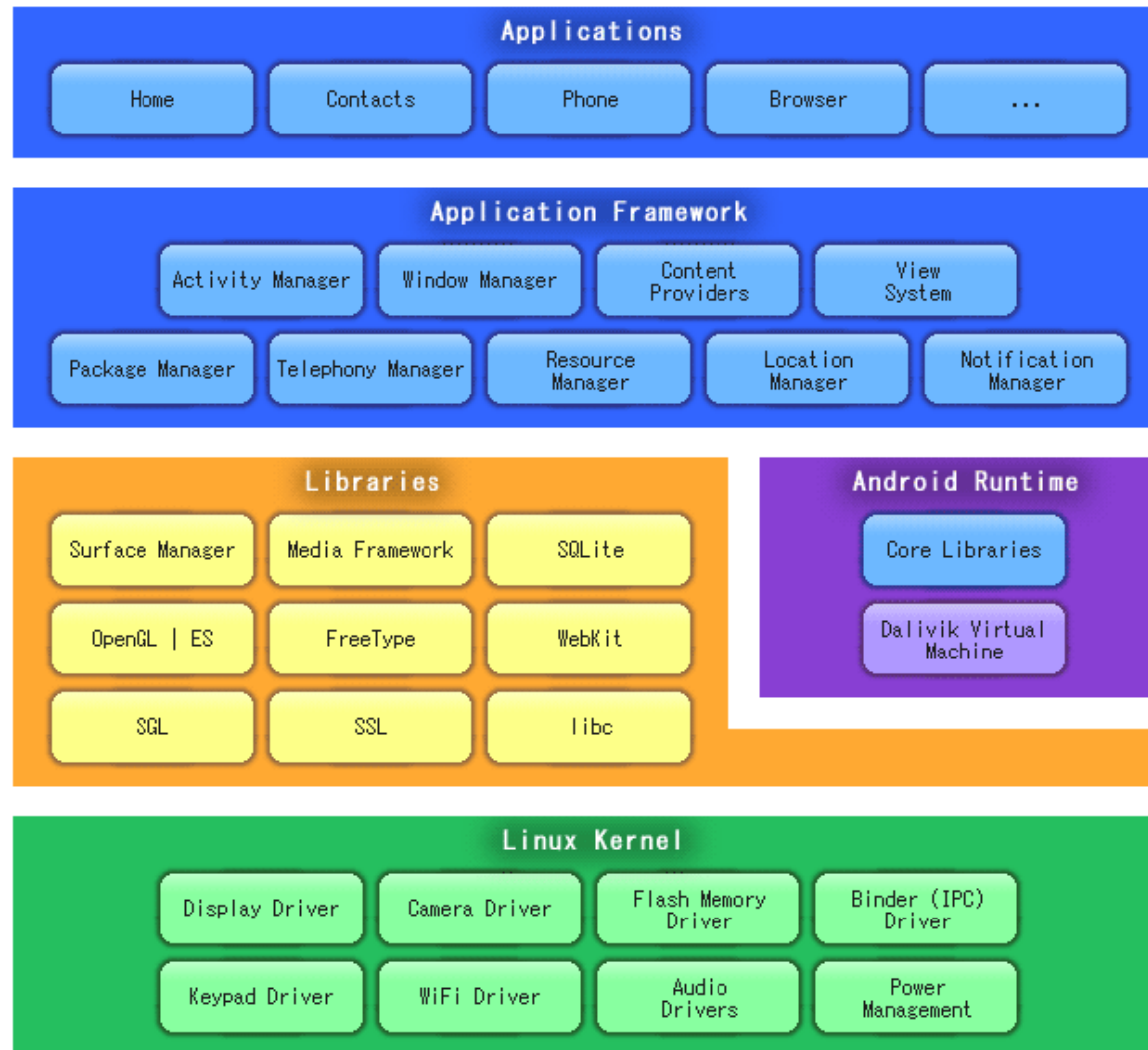




Unvailing Exception Handling **Bug Hazard** in Android based on Github and Google Code Issues

Roberta Coelho|Lucas Almeida|
Georgios Gousios|Arie van Deursen|Natal, Brazil

Androidプラットフォームの構造



背景と紹介

- ❖ Undocumented Exception
Androidにはドキュメントに記載されていない例外が多く存在している
- ❖ 例外判定
例外発生≡スタックトレース*が存在する

```
AndroidRuntime FATAL EXCEPTION: main
AndroidRuntime java.lang.RuntimeException: Unable to start activity ComponentInfo{com.bignerdranch
quiz/com.bignerdranch.android.geoquiz.QuizActivity}: java.lang.NullPointerException
AndroidRuntime at android.app.ActivityThread.performLaunchActivity(ActivityThread.java:2059)
AndroidRuntime at android.app.ActivityThread.handleLaunchActivity(ActivityThread.java:2084)
AndroidRuntime at android.app.ActivityThread.access$600(ActivityThread.java:130)
AndroidRuntime at android.app.ActivityThread$H.handleMessage(ActivityThread.java:1195)
AndroidRuntime at android.os.Handler.dispatchMessage(Handler.java:99)
AndroidRuntime at android.os.Looper.loop(Looper.java:137)
AndroidRuntime at android.app.ActivityThread.main(ActivityThread.java:4745)
AndroidRuntime at java.lang.reflect.Method.invokeNative(Native Method)
AndroidRuntime at java.lang.reflect.Method.invoke(Method.java:511)
AndroidRuntime at com.android.internal.os.ZygoteInit$MethodAndArgsCaller.run(ZygoteInit.java:
AndroidRuntime at com.android.internal.os.ZygoteInit.main(ZygoteInit.java:553)
AndroidRuntime at dalvik.system.NativeStart.main(Native Method)
AndroidRuntime Caused by: java.lang.NullPointerException
AndroidRuntime at com.bignerdranch.android.geoquiz.QuizActivity.updateQuestion(QuizActivity.j
AndroidRuntime at com.bignerdranch.android.geoquiz.QuizActivity.onCreate(QuizActivity.java:90)
AndroidRuntime at android.app.Activity.performCreate(Activity.java:5008)
AndroidRuntime at android.app.Instrumentation.callActivityOnCreate(Instrumentation.java:1079)
AndroidRuntime at android.app.ActivityThread.performLaunchActivity(ActivityThread.java:2023)
AndroidRuntime ... 11 more
ActivityManager Force finishing activity com.bignerdranch.android.geoquiz/.QuizActivity
WindowManager Failure taking screenshot for (246x410) to layer 21010
Choreographer Skipped 31 frames! The application may be doing too much work on its main thread.
ActivityManager Activity pause timeout for ActivityRecord{413bbdc0 com.bignerdranch.android.geoqui
^
```

Androidの例外の問題



exception Android



ウェブ

画像

動画

ニュース

ショッピング

もっと見る ▼

検索ツール

約 29,000,000 件 (0.24 秒)

Exception | Android Developers

[developer.android.com/reference/java/.../Exception.ht...](https://developer.android.com/reference/java/.../Exception.html) ▼ このページを訳す

Android Developers. **Exception**. close. Developer Console. Results. Loading...

Android Developers public class. **Exception**. extends Throwable · java.lang.
Object. ↳, java.lang.Throwable. ↳, java.lang.**Exception**. Known Direct
Subclasses.

[Throwable](#) - [RuntimeException](#) - [NullPointerException](#)

- [IllegalArgumentException](#)

Androidアプリ開発で例外の発生した場所を特定する ... **77 users**

dev.classmethod.jp/smartphone/android/android-app-exception/ ▼

2013/02/18 - この記事は公開されてから1年以上経過しています。情報が古い可能性がありますので、ご注意ください。以前、「iOSアプリ開発で例外の発生した場所を特定する」というブログを書きましたが、**Android**でも同じような機能があるようです。

例外の分類

例外を次の4つに分類

1. APIのドキュメントに記載されているがcatchできていない例外
→開発者の不注意
2. リソース不足によって起こる例外
3. 親クラスの例外でCatchしてしまっている例外
→NullPointerExceptionをExceptionクラスでcatchする
4. コードの再利用による例外
→throw推奨の例外をthrowしている箇所を再利用することで、throwすべきでない例外までthrowしてしまう

Study Design

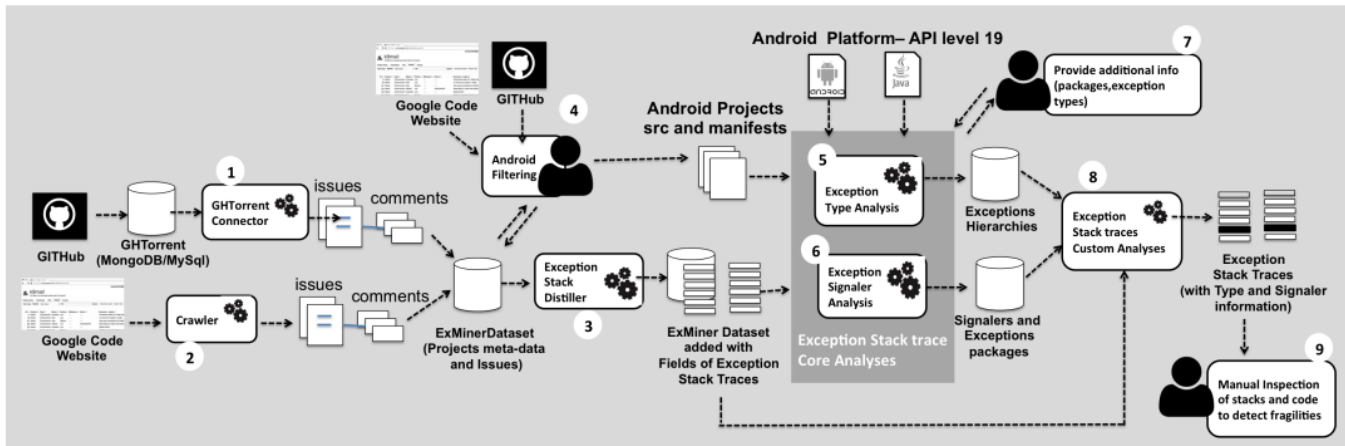


Fig. 2: Study overview.

482のAndroidOSS、31592の不具合、4042のstack trace

- ❖ GitとGoogleCodeIssuesの不具合報告を集める
- ❖ スタックトレースを不具合とリンクする
- ❖ ソースコード、バイトコードに分けて分析を行う
- ❖ 例外を分類し、その原因を探る

結果 —例外の原因を探る

Root Exception	Projects		Occurrences		Android	Libcore	App	Lib
	#	%	#	%				
java.lang.NullPointerException	332	51.96%	1664	27.71%	525	20	836	280
java.lang.IllegalStateException	120	18.78%	278	4.63%	185	31	41	39
java.lang.IllegalArgumentException	142	22.22%	353	5.88%	195	12	95	44
java.lang.RuntimeException	122	19.09%	319	5.31%	203	2	64	51
java.lang.OutOfMemoryError	78	12.21%	237	3.95%	141	16	35	34
java.lang.NoClassDefFoundError	67	10.49%	94	1.57%	10	0	46	37
java.lang.ClassCastException	64	10.02%	130	2.16%	55	0	55	20
java.lang.IndexOutOfBoundsException	62	9.70%	166	2.76%	53	0	93	18
java.lang.NoSuchMethodError	54	8.45%	80	1.33%	10	0	56	14
java.util.ConcurrentModificationException	43	6.73%	65	1.08%	5	0	46	13

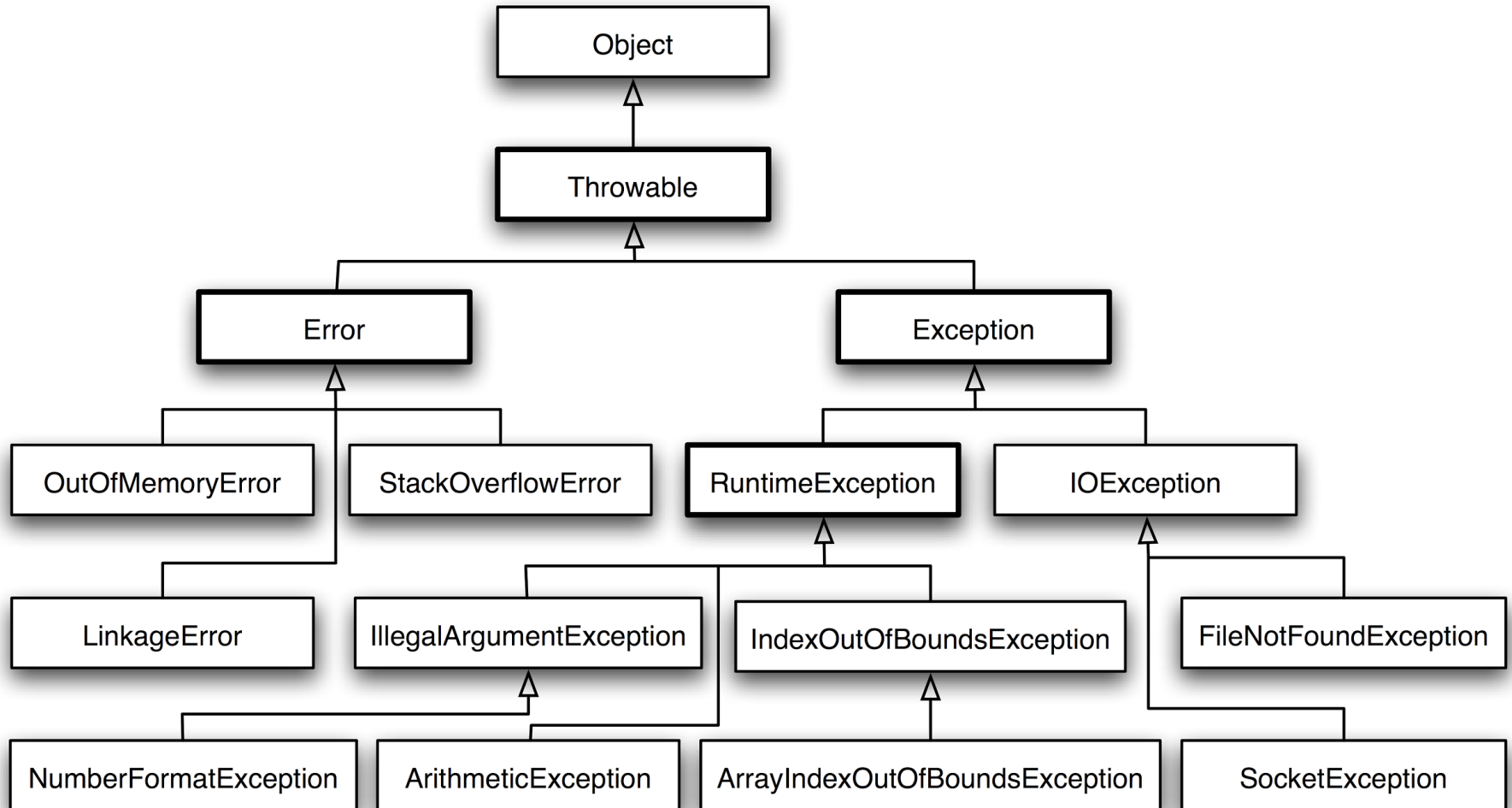
TABLE III: Root Exceptions occurrences and popularity in repositories hosted in Google Code (GC) and GitHub(GH).

Concern	% Occurrences on stacks
Programming logic (java.lang and util)	52,0%
Resources (IO, Memory, Batory)	23,9%
Security	4,1%
Concurrency	2,9%
Backward compatibility	5,5%
Specific Exceptions	4,9%
General (Error, Exception, Runtime)	6,7%

TABLE IV: Identifying the concerns related to root exceptions

- ❖ 1/2がプログラミングによる例外(NPE, OOB)
- ❖ 1/4がリソース不足による例外(IO,Memory)

結果 —例外の種類を探る



結果 —例外の種類を探る

Root Type	Android	Libcore	App	Lib	All	%
Runtime	1335	73	1843	690	3894	64.85%
Error	188	46	302	167	691	11.51%
Checked	276	314	313	567	1358	22.61%
Throwable	0	0	2	0	2	0.03%
Undefined	4	0	18	38	60	1.00%
All	1 803	433	2478	1462	6005	

TABLE V: Types and origins of root exceptions.

❖ 2/3がRuntime(アプリケーション層)の例外

Origin	stacks	signaler methods	throws clause	@throws
Libraries	205	44	0	1
Android	62	35	0	0
All	267	79	0	1

TABLE VI: Absence of exception interfaces on methods.

Only a small fraction (4%, 267 stack traces) of run time exceptions are programmatically thrown. Almost none (0.4%, just one) of these were documented.

結果 —Bug Hazardについて

Wrapper Type	Root Cause Type	Projects	Occurrences	Android	Java/Libcore	Lib	App
Runtime	Checked	88	148	75	0	38	35
Runtime	Error	46	67	58	0	8	1
Checked	Runtime	17	31	4	0	16	11
Checked	Error	8	9	5	0	1	3
Error	Checked	14	27	6	7	6	8
Error	Runtime	8	17	1	1	1	14

TABLE VIII: Wrappings comprising different exception types.

- ❖ Cross-typeな例外が多く見られた(50.7%)
→Aで起こった例外がBの例外を引き起こす
- ❖ これらはバグチェーン(Bug Hazard)に繋がる

まとめ

- ❖ Androidプラットフォームの問題点を指摘
- ❖ 例外の原因を詳細に調査
 - 例外の1/2が開発者に起因
 - 1/4がリソースの不足に起因
- ❖ 例外の種類を調査
 - 2/3がRuntime(アプリケーション層)の例外
- ❖ Bug Hazardについて調査
 - 1/2の例外がCross-Typeな例外
 - これらはBug Hazardを引き起こす危険性を孕んでいる