

アンドロイドアプリにおけるバグ報告の品質を高めることを目指す

Auto-completing Bug Reports for Android Applications

Auto-complete

入力履歴を活用して入力操作の軽減を図る機能の1つ

Kevin Moran

Mario Linares-Vásquez

Carlos Bernal-Cárdenas

Denys Poshyvanyk

B4 作島 大智

目次

- ❖ 背景
- ❖ 問題点
- ❖ 提案手法について
- ❖ 提案手法の評価実験

背景

- ❖ スマホやモバイルコンピュータが普及している
- ❖ バグ報告が非構造的で理解しづらい
- ❖ GUIのバグ報告で再現手順を報告することが困難
- ❖ モバイルアプリケーションの開発と保守が困難

関連研究

- ❖ 面倒で多くの報告者は詳細な説明をしない [23,34,22,15]
- ❖ 専門家と非専門家ではバグ報告の記述に差がある [40]
- ❖ 情報の欠如が再現性のないバグ報告の主な原因 [35,21]

課題点

- ❖ 報告者が再現できない報告をする
 - ❖ バグ報告をもとに開発者が修正できない
 - ❖ 開発者が望む情報と報告内容のギャップを埋めたい
- 開発者がバグを再現する際に高レベルな機能説明からコードレベルの問題について推論する必要がある

提案手法

開発者が理解できるバグ報告を報告者が作成することを支援する

品質の高いバグ報告

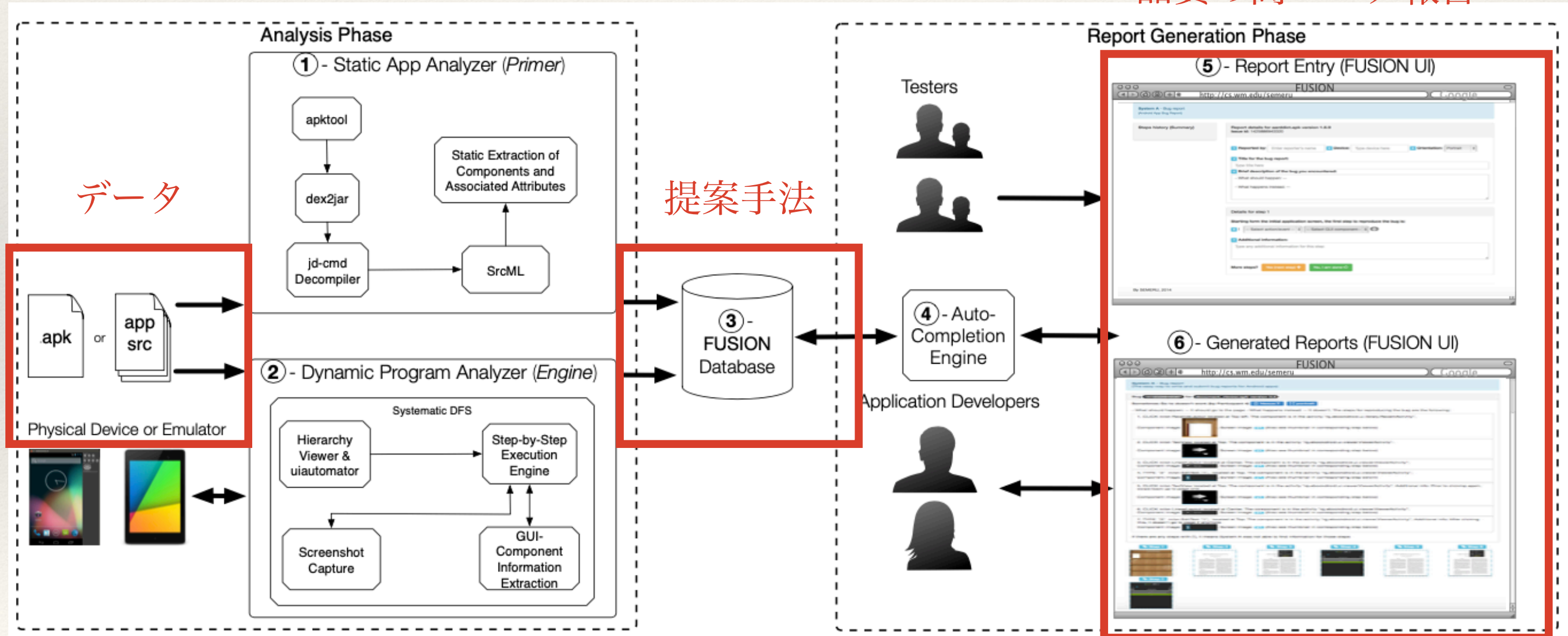


Figure 1: Overview of FUSION Workflow

①静的解析

静的解析

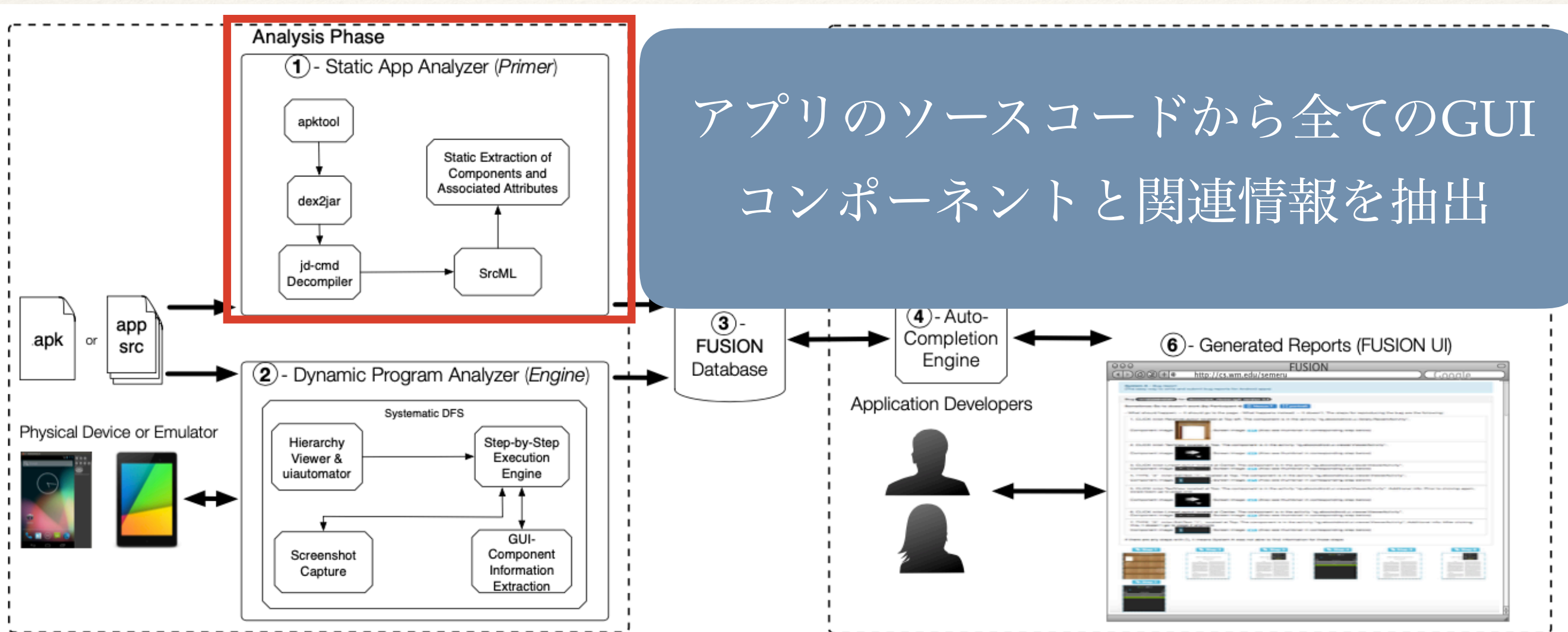


Figure 1: Overview of FUSION Workflow

- ❖ GUIコンポーネント：タップ、ロングタッチ、スワイプ、タイプなど

①静的解析

Analysis Phase

① - Static App Analyzer (*Primer*)

App APKから
リソースファイルを
抽出

apktool [2]

dex2jar [4]

デコンパイル

jd-cmd
Decompiler [8]

Static Extraction of
Components and
Associated Attributes

SrcML [12]

ソースファイルを
XMLベースの表現
に変換

②動的解析

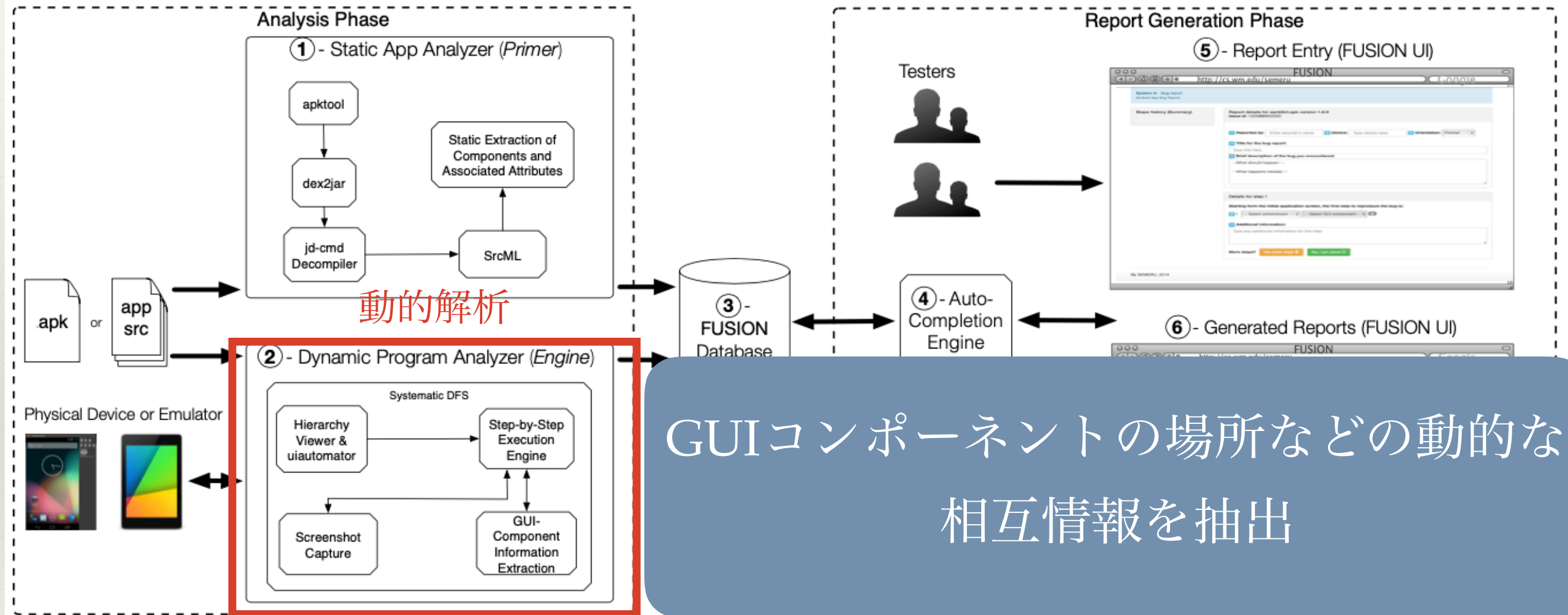
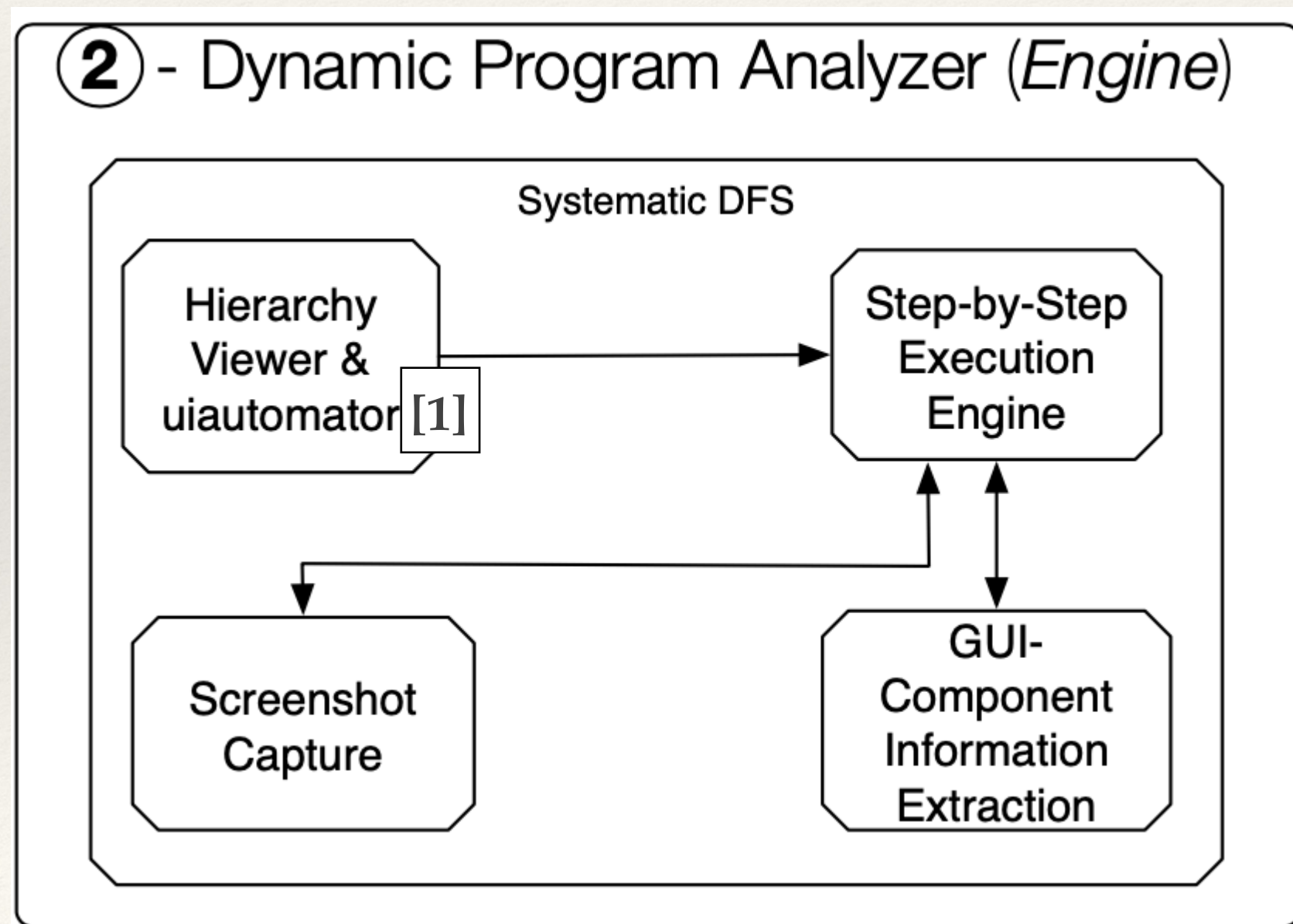


Figure 1: Overview of FUSION Workflow

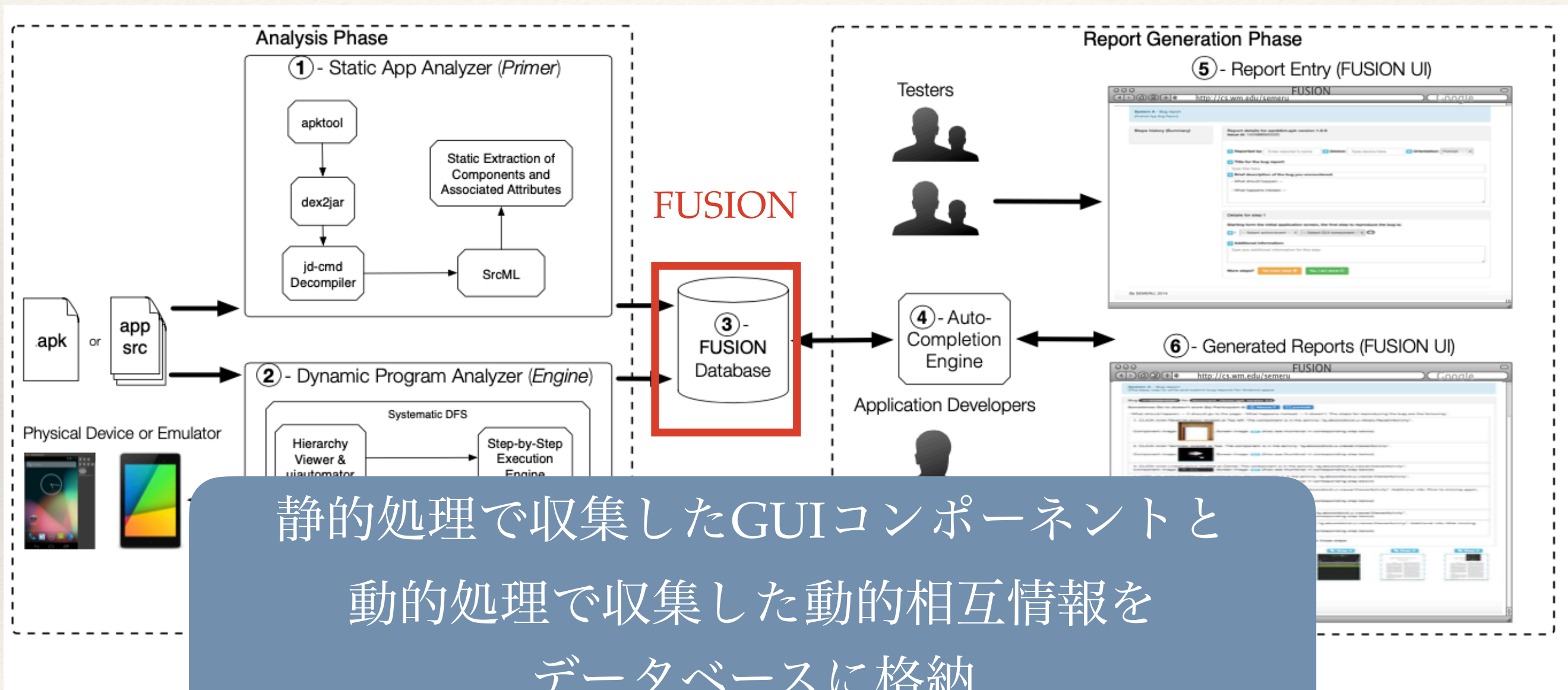
- ❖ 送信などのGUIコンポーネントに関連付けられたテキストや各アクションが実行される前後のフルスクリーンショットなど

②動的解析

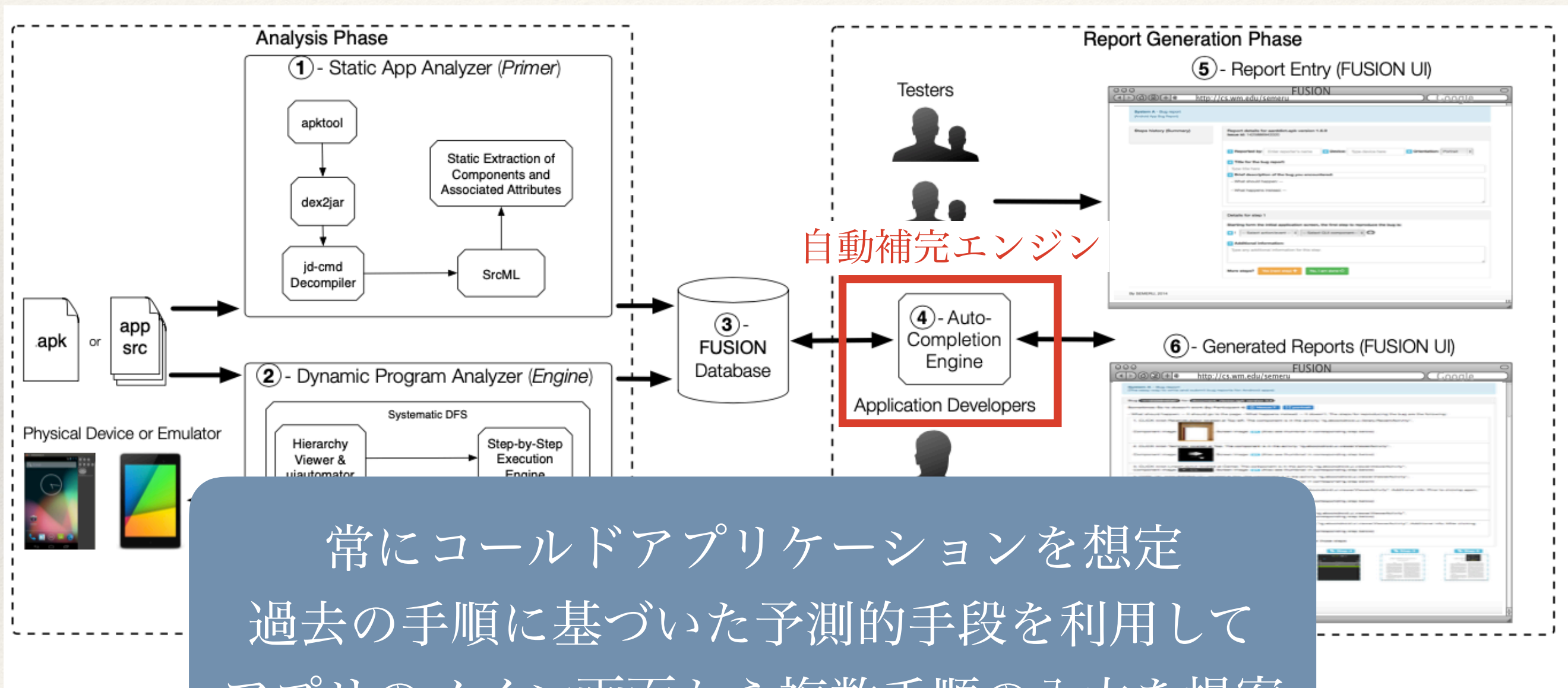
- ・アプリケーションを効果的に調査するために深さ優先探索(DFS)を使用
- ・Android SDKに含まれるUIAutomator [1]フレームワークを使用する



③FUSION



④ Auto-Completion Engine



常にコールドアプリケーションを想定
過去の手順に基づいた予測的手段を利用して
アプリのメイン画面から複数手順の入力を提案

⑤報告入力例

Report details for document_viewer.apk version 2.2

Issue id: 1426467887725

1

Reported by:

Reporter

2

Device:

Nexus 7

3

Orientation:

Portrait

4

Title for the bug report:

The GoTo Page feature does not work Properly

5

Brief description of the bug you encountered:

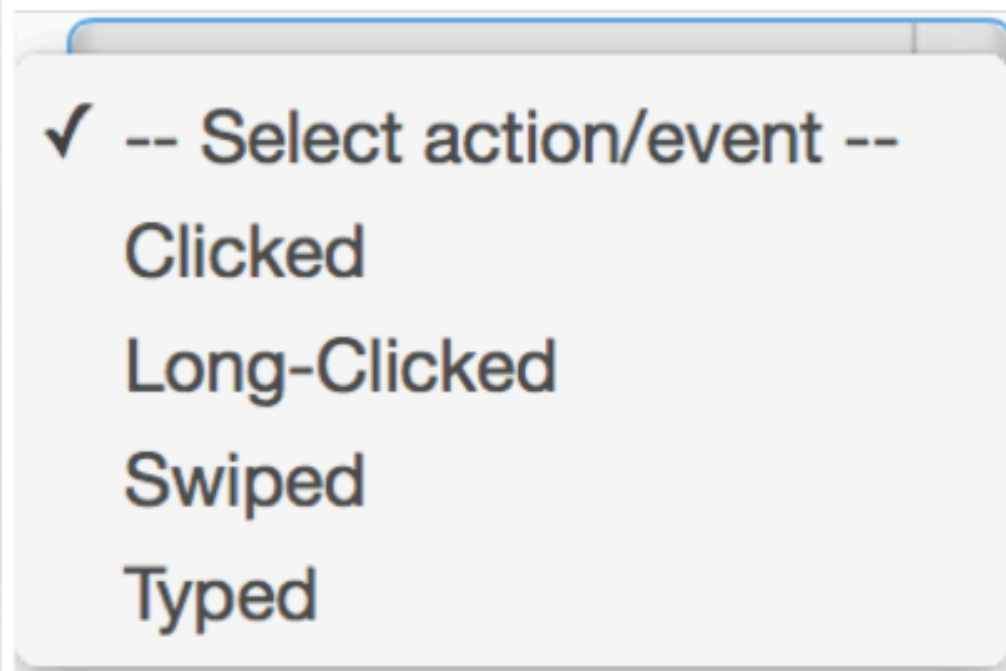
- What should happen: --
Tapping the GoTo button brings you to the corresponding page

- What happens instead: --
You stay on the same page.

Figure 2: FUSION Reporter Interface

ドロップダウンリスト

A) Action List



B) Component List

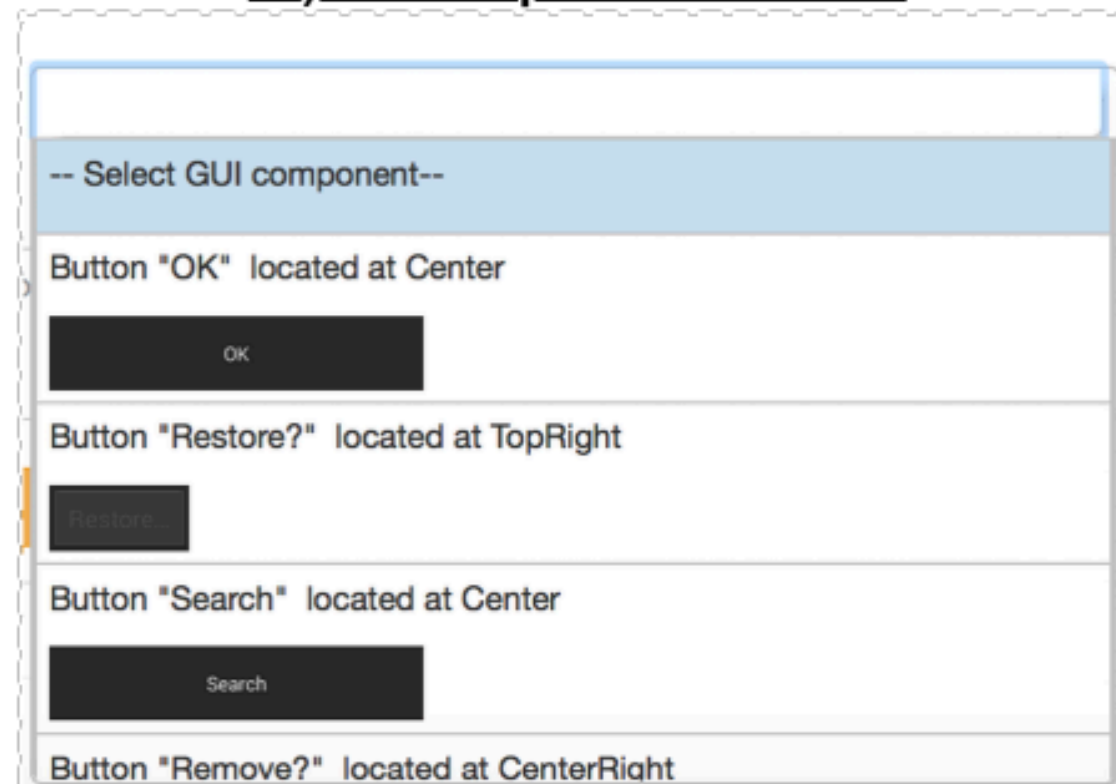


Figure 3: Auto-Complete Dropdown Menus

⑥バグ報告の構造

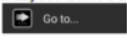
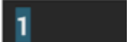
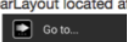
手順



Bug 1418328948387 for document_viewer.apk version 2.2

Sometimes Go to doesn't work (by Participant 4) Nexus 7 portrait

- What should happen: -- It should go to the page - What happens instead: -- It doesn't. The steps for reproducing the bug are the following:

1. CLICK in/on RelativeLayout located at Top left. The component is in the activity "rg.ebookroid.ui.library.RecentActivity".
Component image:  Screen Image:  (Also see thumbnail in corresponding step below)
2. CLICK in/on TextView located at Top. The component is in the activity "rg.ebookroid.ui.viewer.ViewerActivity".
Component image:  Screen Image:  (Also see thumbnail in corresponding step below)
3. CLICK in/on LinearLayout located at Center. The component is in the activity "rg.ebookroid.ui.viewer.ViewerActivity".
Component image:  Screen Image:  (Also see thumbnail in corresponding step below)
4. TYPE "2" in/on EditText "1", located at Top. The component is in the activity "rg.ebookroid.ui.viewer.ViewerActivity".
Component image:  Screen Image:  (Also see thumbnail in corresponding step below)
5. CLICK in/on TextView located at Top. The component is in the activity "rg.ebookroid.ui.viewer.ViewerActivity". Additional info: Prior to clicking again, swipe back up to page one
Component image:  Screen Image:  (Also see thumbnail in corresponding step below)
6. CLICK in/on LinearLayout located at Center. The component is in the activity "rg.ebookroid.ui.viewer.ViewerActivity".
Component image:  Screen Image:  (Also see thumbnail in corresponding step below)
7. TYPE "2" in/on EditText "1", located at Top. The component is in the activity "rg.ebookroid.ui.viewer.ViewerActivity". Additional info: After clicking this, it doesn't go to page 2 anymore
Component image:  Screen Image:  (Also see thumbnail in corresponding step below)

If there are any steps with (*), it means System A was not able to find information for those steps

Step 1 Step 2 Step 3 Step 4 Step 5 Step 6



報告のタイトル, デバイス,
短い説明などの予備説明

- (i)各ステップのアクション
- (ii)コンポーネントのタイプ
- (iii)相対位置コンポーネント
- (iv)コンポーネント固有
のスクリーンショット

各ステップの
スクリーンショットのリスト

リサーチクエスチョン

- ❖ RQ1：Androidアプリのバグを報告する場合、バグレポートのどの情報フィールドが役立つのか。
- ❖ RQ2：FUSIONは従来のバグ追跡システムよりもバグの報告に使いやすいか。
- ❖ RQ3：FUSIONは従来のバグレポートよりもバグの再現に使いやすいか。
- ❖ RQ4：FUSIONで生成されたバグレポートでは、従来のバグ追跡システムを使用して送信されたレポートと比較して、バグを迅速に再現できるか。
- ❖ RQ5：FUSIONを使用する開発者は、従来のバグ追跡システムと比較して、より多くのバグを再現できるのか。

リサーチクエスチョン

- ❖ RQ1：Androidアプリのバグを報告する場合、バグレポートのどの情報フィールドが役立つのか。
 - ❖ RQ2：FUSIONは従来のバグ追跡システムよりもバグの報告に使いやすいか。
 - ❖ RQ3：FUSIONは従来のバグ追跡システムよりもバグの報告に使いやすいか。
 - ❖ RQ4：FUSIONは従来のバグ追跡システムよりもバグの報告に使いやすいか。
 - ❖ RQ5：FUSIONは従来のバグ追跡システムよりもバグの報告に使いやすいか。
- RQ1・RQ2
- FUSIONの機能が有効かどうかを評価する
- ❖ RQ6：FUSIONは従来のバグ追跡システムよりもバグの報告に使いやすいか。

リサーチクエスチョン

RQ3・RQ4・RQ5

- ❖ RQ3：FUSIONを使用してバグを報告する開発者の使いやすさと好みを評価することと
情報
- ❖ RQ4：FUSIONで生成されたバグレポートでは、従来のバグ追跡システムを使用して送信されたレポートと比較して、バグを迅速に再現できるか。
いか。
- ❖ RQ5：FUSIONを使用する開発者は、従来のバグ追跡システムと比較して、より多くのバグを再現できるのか。

評価実験

データセット

- ❖ GUIの観点からコンテキストに依存しないバグ
- ❖ さまざまな複雑さのバグ
- ❖ Nexus 7タブレットで再現可能なバグ

比較対象

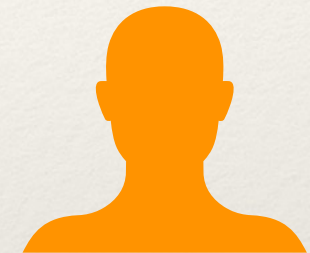
- ❖ Google Code Issue Tracker (GCIT)

RQ1：有用な情報フィールド

経験豊富な開発者



経験の浅い開発者



情報を入力するだけで楽

再現手順の各ステップを効果的に報告するのに
GUIコンポーネントフォームとアクション/イベントフォームが有効

- ❖ 追加情報を記入する欄を設けることで報告の品質を高めるが、経験が浅い開発者はシンプルなWeb GUIが好まれる

RQ2：提案手法の使いやすさ

経験豊富な開発者

経験の浅い開発者

	Participant #1 (EX)	Participant #2 (EX)	Participant #3 (IEX)	Participant #4 (IEX)
FUSION:	5:14	5:20	10:40	4:59
	Participant #5 (EX)	Participant #6 (EX)	Participant #7 (IEX)	Participant #8 (IEX)
GCIT:	3:17	6:39	1:14	1:46

- ❖ 経験豊富な開発者は両方同様に扱えるが
経験が浅い開発者には使用が困難

RQ3：バグ報告の使いやすさ

- ❖ UX-related：User Experienceに基づく質問 [26]

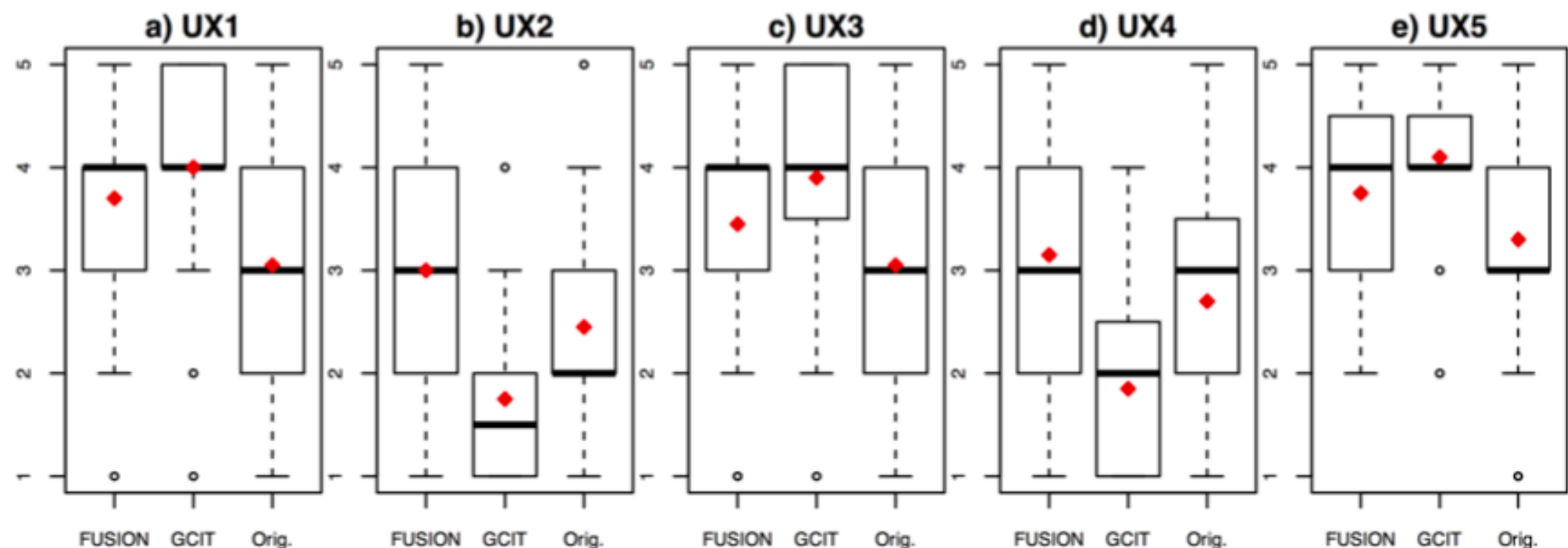


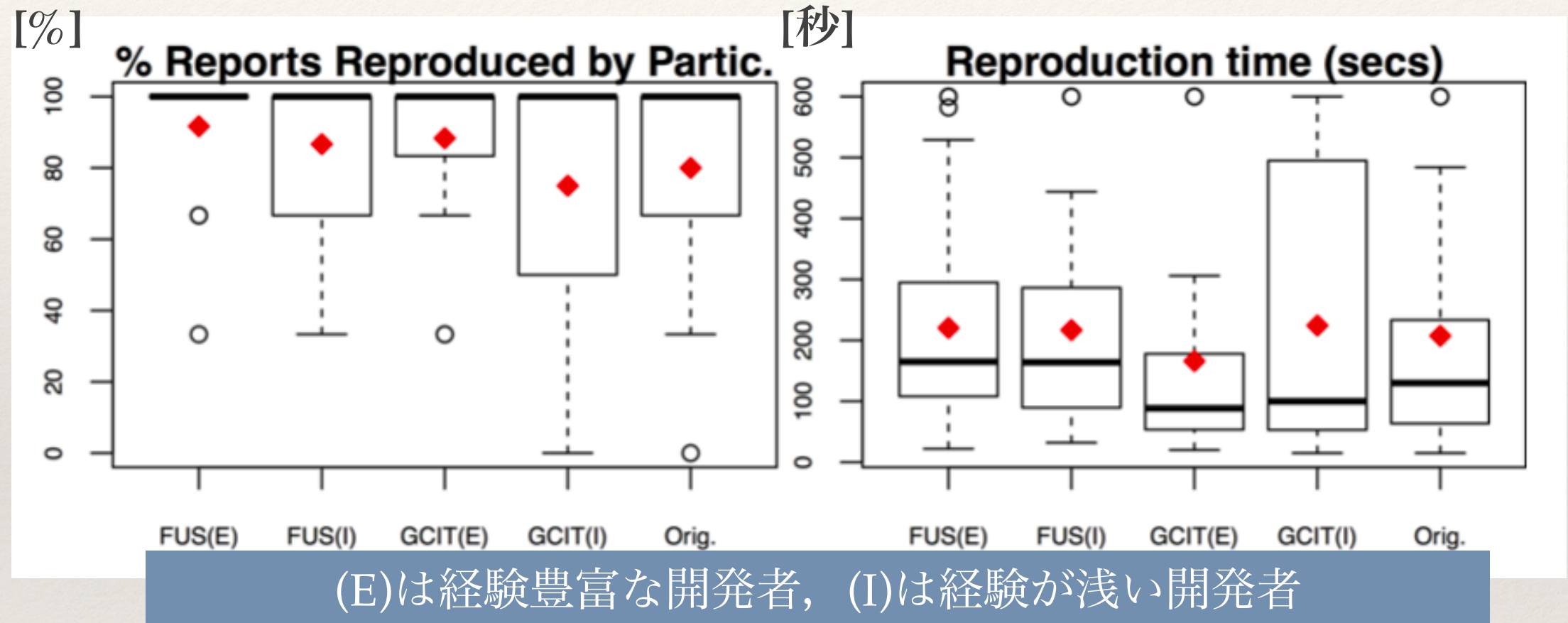
Figure 7: Answers to the UX-related questions in RQ₃

- ❖ 従来手法のほうがわずかにユーザビリティスコアが高い
- ❖ 場合によっては手順が詳細すぎることを原因と考えられる

RQ4：再現にかかる時間

個人が再現できたバグの割合

個人が再現にかかった時間



- ❖ 提案手法は従来手法(GCIT)と比較して、バグを迅速に再現することはできない

RQ5：再現できるバグの数

手法	開発者	再現	非再現	割合
FUSION	経験豊富	55	5	92%
FUSION	経験が浅い	52	8	87%
GCIT	経験豊富	52	8	87%
GCIT	経験が浅い	45	15	75%

- ❖ 経験が浅い開発者にとってバグ報告の品質を大きく改善できることを意味する
- ❖ 提案手法は従来手法(GCIT)と比較してより多くのバグを再現できる

制限と妥当性への脅威

制限

- ❖ タップ、ロングタッチ、スワイプ、タイプなどのGUIジェスチャーのみを使用して発見できる機能的なバグのみを対象としている
- ❖ 微妙なパフォーマンスのバグやコンテキスト依存のバグには適していないバグが存在する

妥当性への脅威

- ❖ F-Droid [5]マーケットプレイスの異なる14のアプリケーションからの15のバグレポートのみで評価していること

まとめ

- ❖ バグの再現手順を自動完了する支援するためにAndroidアプリケーションのイベント駆動型の性質を活用する新しいバグ報告アプローチであるFUSIONを提案
- ❖ 包括的な評価結果は、提案手法は再現性の高いバグ報告を生成することができる