

Crowdsourcing을 이용한 안드로이드 플랫폼 라이브러리 정보 수집

박지순 이광근

서울대학교 프로그래밍 연구실

1 동기

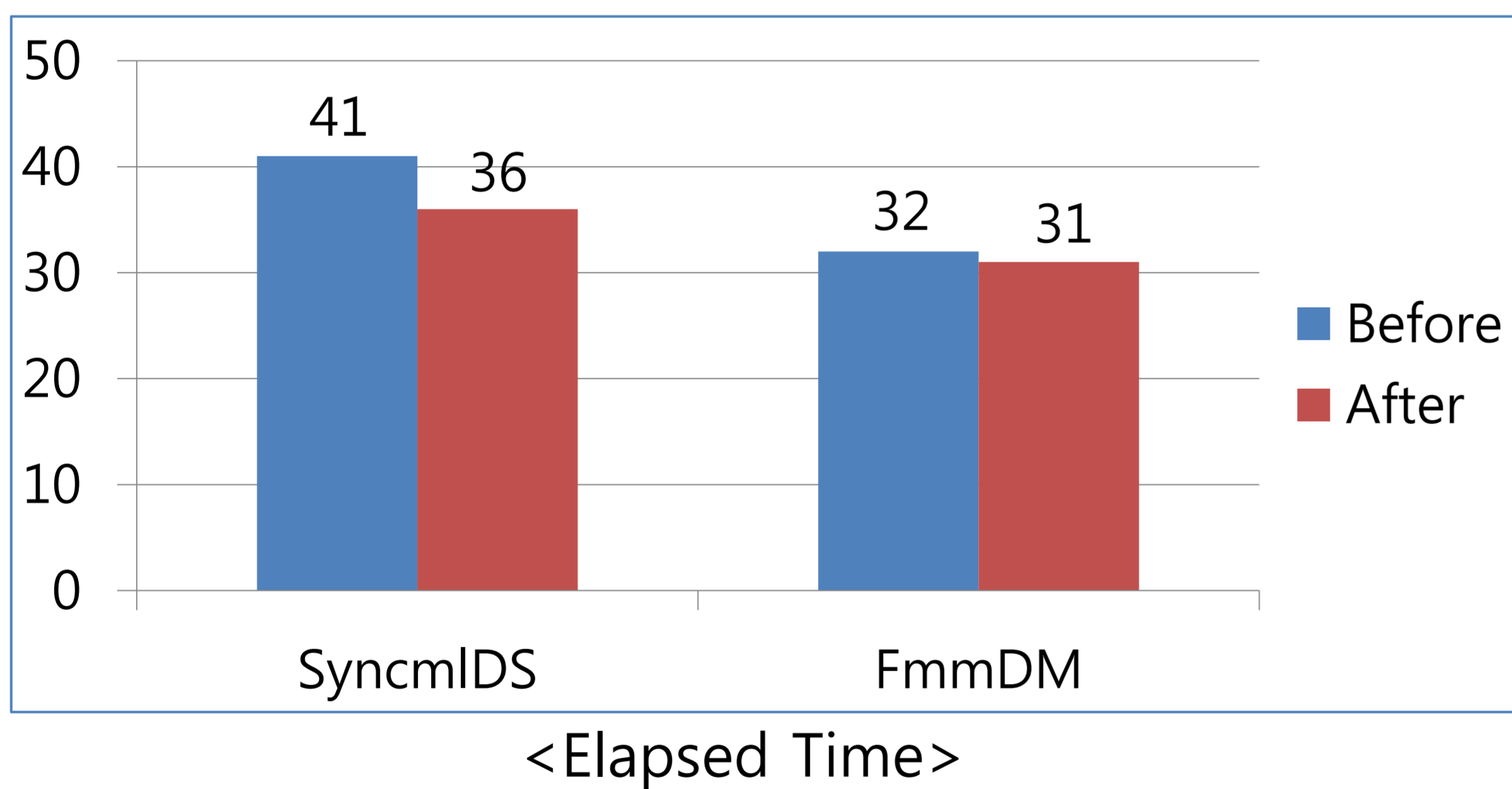
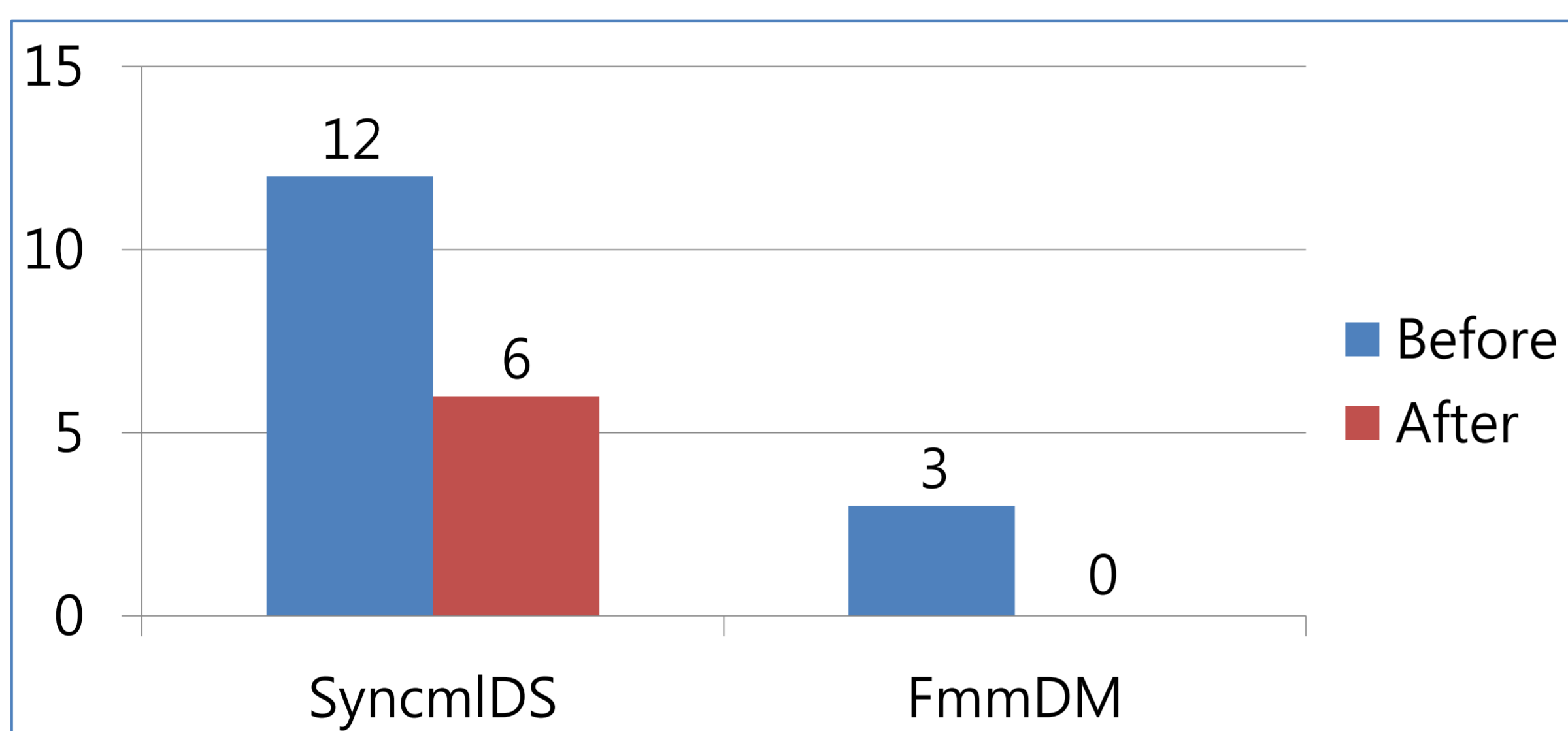
- 안전한(Sound) Android App 정적 분석기 **ScanDal**
- 경우에 따라 충분한 정확성을 제공하지 못하는 경우 발생
 - apk에 없는 플랫폼 라이브러리 호출 때문
 - 허위경보 (False positive) 발생, 분석 성능 하락
- 라이브러리 분석기 구현은 복잡하고 시간이 오래 걸림

API를 직관적으로 이해할 수 있는
"사람의 이해력"을 이용해 보자!!

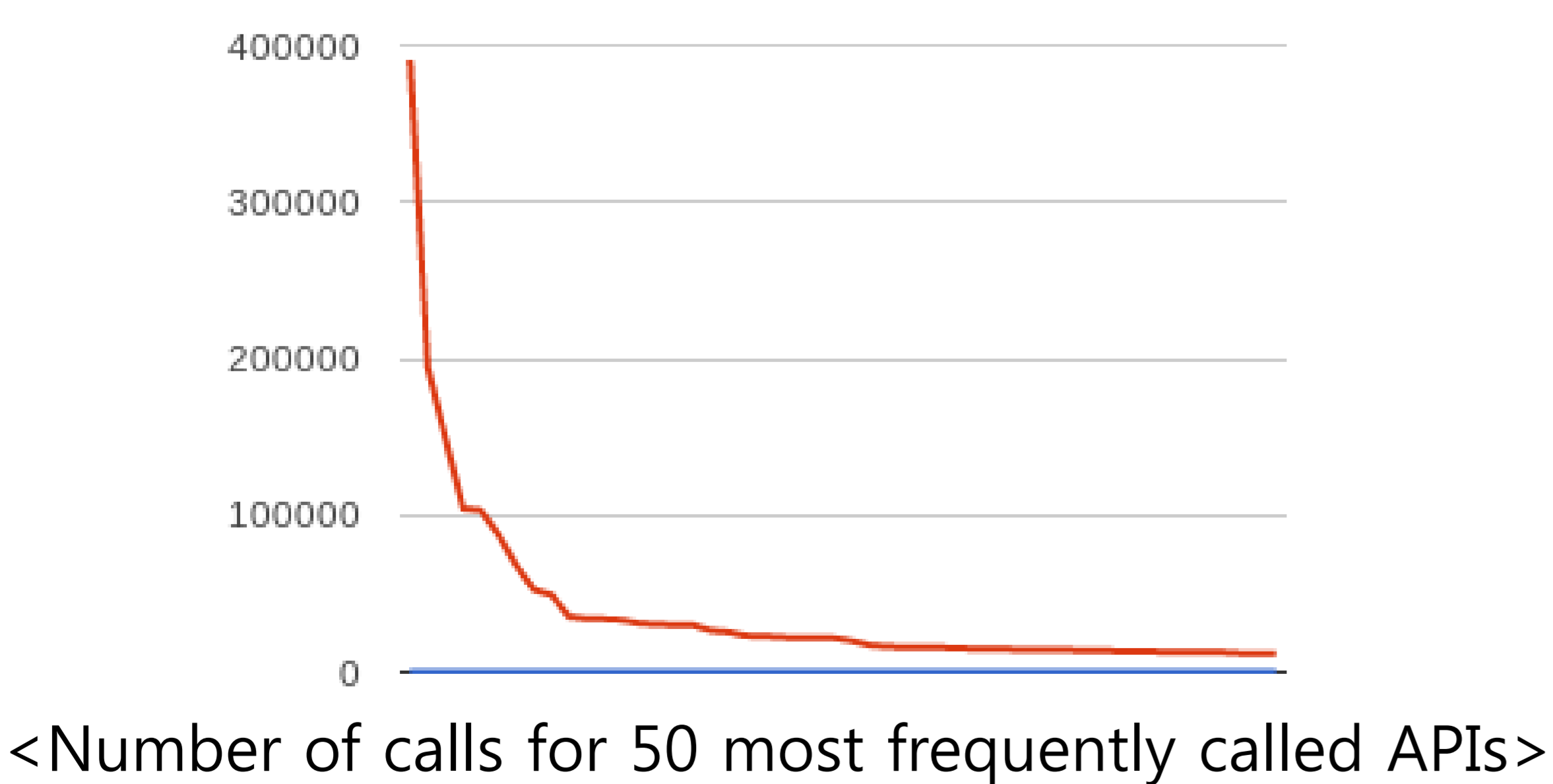


2 들여다보기

- 필요한 정보는?
 - API 호출에 따른 데이터의 전파 정보
- 데이터 전파 정보를 알게 된다면?!

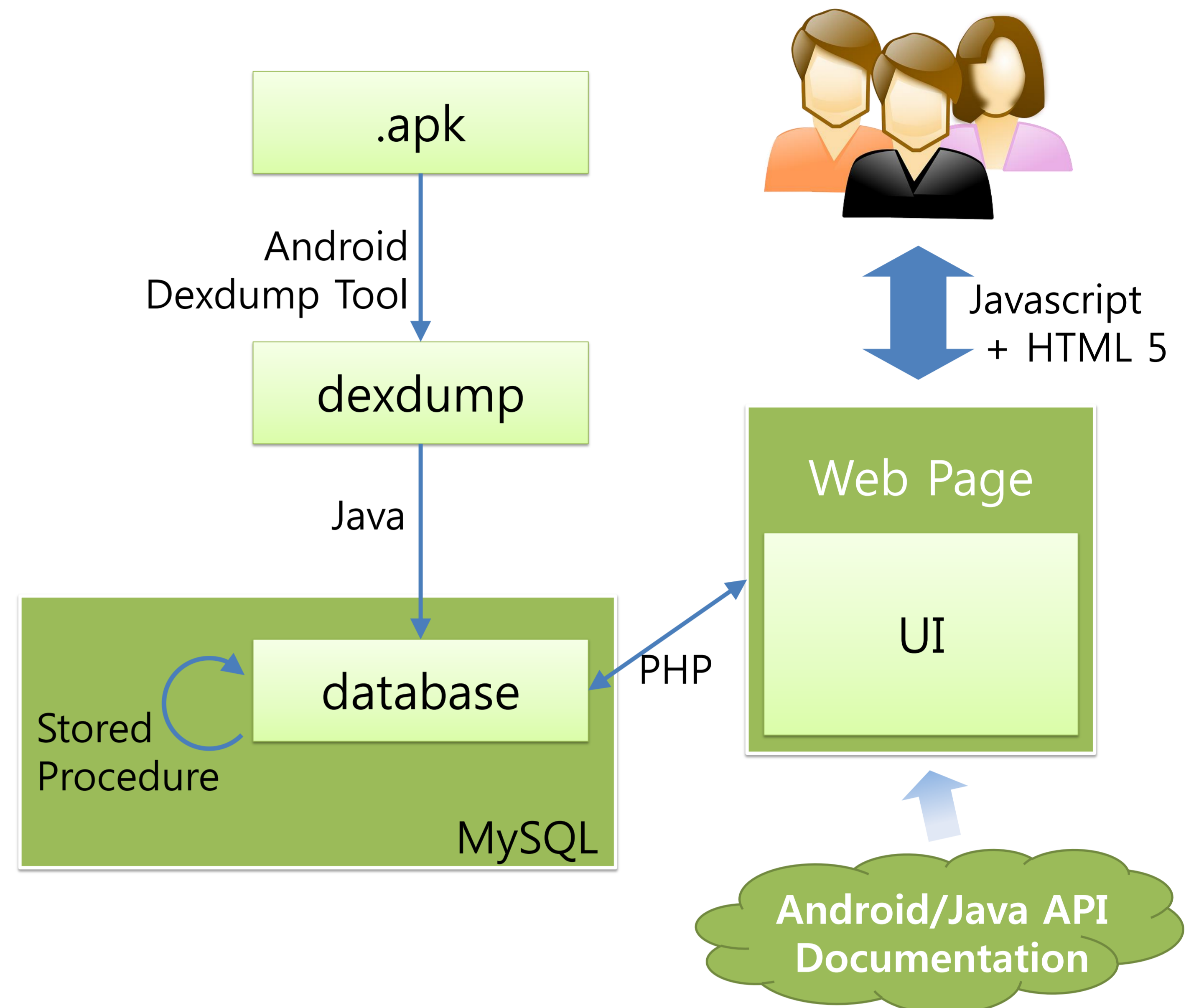


- 플랫폼 라이브러리 API 호출 특성
 - 자주 호출되는 10%의 API가 전체 호출의 93%를 차지
 - 분석할 API에 우선순위를 부여



3 DroidPang

- DroidPang 프로젝트의 구성



4 API 분석 UI

- 분석 페이지 구성
 - reCAPTCHA의 아이디어
 - . 한번에 2개씩 풀기
 - . 첫 번째 문제를 제대로 못 풀면 두 번째 문제의 결과는 버림
 - API에 대한 설명을 읽고
 1. 전파 경로를 따라 줄긋기
 2. Control Flow를 바꾸는 API 식별

Class: java.lang.String
Method: substring()

```

    public String substring(int beginIndex)
    Returns a new string that is a substring of this string. The substring begins with the character at the specified index and extends to the end of this string.
    Examples:
    "unhappy".substring(2) returns "happy"
    "Harbison".substring(3) returns "bison"
    "emptiness".substring(9) returns "" (an empty string)
    Parameters:
    beginIndex - the beginning index, inclusive.
    Returns:
    the specified substring.
    Throws:
    
```

Legend:

- this
- beginIndex
- return value

Legend:

- ☐ This function creates a thread
- ☐ This function takes callback function or event handler as an argument

- Project URL
<http://ropas.snu.ac.kr/droidpang>