

# MOBILE app Data collection and Analysis (MODAL): An Infrastructure for Improving Mobile App Quality

Wing Lam  
George Mason University  
winglam@gmu.edu

Kevin Moran  
University of Central Florida  
kpmoran@ucf.edu

Wei Yang  
University of Texas at Dallas  
wei.yang@utdallas

## Objective

This project outlines a strategy to create approaches that lower the hurdles to high-impact research aimed at enhancing mobile app quality. To do so, we plan to organize planning activities to identify current challenges in mobile app research and to design a community infrastructure that can help researchers overcome these challenges.

## Approaches

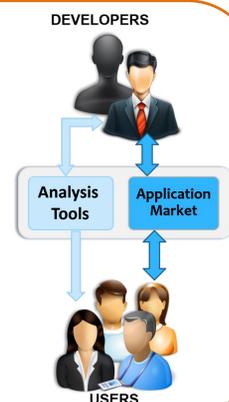
- **Data Detection** – Crawling apps from app stores and integrating with continuous-integration (CI) pipelines to automatically download and analyze code changes.
- **Data Mining** – Mining both static and dynamic app information every time an app updates. The infrastructure includes a set of automated app navigation tools.

## Mobile App Data Collection and Analysis

**Reliance on Data and Analysis Tools** - Research in mobile app quality and development practices heavily relies on two types of artifacts: static and dynamic app data, and analysis techniques and tools.

**Past Challenges with Ad-Hoc Tools and Data** - Historically, research in this area has suffered from a lack of reproducibility and replicability due to the ad-hoc creation of tools and datasets.

**Vision** - A robust infrastructure could provide up-to-date, large-scale datasets and analysis tools. It would be a centralized solution to overcome the barriers of rapid evolution, scale, and tool reliability.



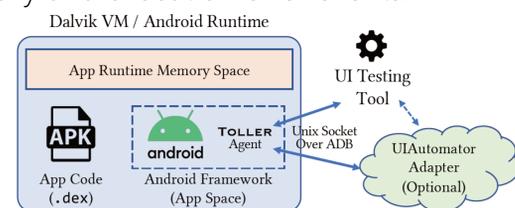
## PI's Prior Work: TOLLER: Infrastructure enhancement to Mobile OS

**Motivation** - Existing app navigating tools, infrastructure-supported operations, such as UI Event Execution, are time-consuming, using up to 70% of the total runtime.

**Toller [1]** - Enhances App Navigation by directly accessing an app's UI data structures and memory, which allows for faster extraction of UI hierarchy and execution of UI events.

**Result** – Reduce operations from several sec. to ms.

**Use Cases** – Enable fast data collection and dynamic analysis by incorporating Toller with other tools.



## Effectiveness and Efficiency of Data Collection

- Toller reduces average time usage of UI Hierarchy Capturing and UI Event Execution operations by up to 97% and 95%, respectively. We integrate Toller with existing App navigation tools and achieve the range of 11.8% to 70.1% relative code coverage improvement on average.
- Our EPIT approach helps Monkey, Ape, and WCTest reach comparable code coverage using 65.9%, 50.1%, and 65.9% less machine time, and 64.0%, 48%, and 41.0% less duration, respectively. When given the same amount of machine time, on avg. EPIT helps Monkey and WCTest (two state-of-the-practice tools) achieve 20.4%, and 10.2% more code coverage.

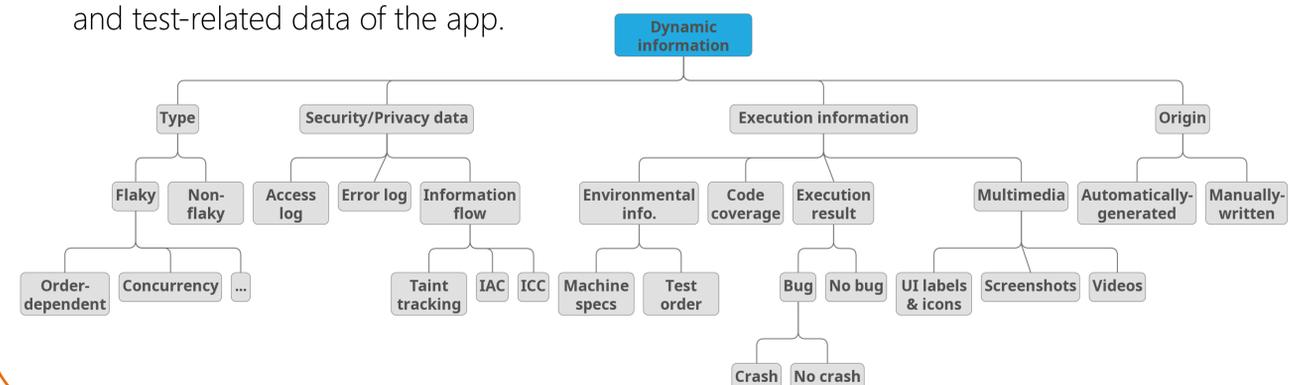
## Current Work: Dynamic Information Collection

### Automated Information Collection

- Dynamic Information is collected by running tools/tests whenever an app updates – recording execution information, security and privacy data, and test-related data of the app.

### Example: Information-flow

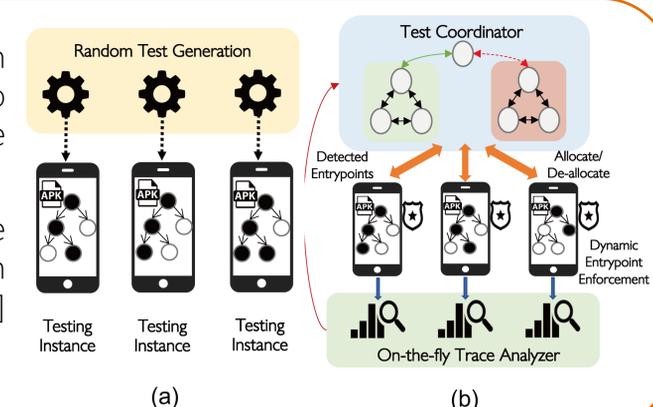
- Notify users of potential information leak by presenting information flows that show what private data flows to what output channels
- Allow users to inspect this information at runtime



## Current Work: Parallelized Data Collection

**Motivation** - Our infrastructure will use an effective parallelized data collection strategy so that each data collection device will collect unique data to better utilize machine resources.

**EPIT** - a fully automated approach for effective parallelization applicable for any app navigation tool based on loosely-coupled UI subspaces [2] that can be identified on-the-fly.



### Reference:

- [1] Wenyu Wang, Wing Lam, and Tao Xie. "An Infrastructure Approach to Improving Effectiveness of Android UI Testing Tools." *In ISSSTA 2021*.
- [2] Wenyu Wang, Wei Yang, Tianyin Xu, and Tao Xie. "VET: Identifying and Avoiding UI Exploration Tarpits." *In ESEC/FSE 2021*.