

Flux: Multi-Surface Computing in Android

EuroSys'15

201924169 Ju-Won Park

June 13, 2019

Problem statement

- There is little system support for multi-surface apps.
- Yet general systems support for multi-surface app is limited to (1) screencasting, which relies on a single master devices computing power and battery life or (2) cloud backing, which is unsuitable in the face of disconnected operation or untrusted cloud providers.

Solution approach

- The authors present an alternative approach: Flux, an Android-based system that enables any app to become multi-surface through app migration
- Selective Record/Adaptive Replay records just those device-agnostic app calls that lead to the generation of app-specific device-dependent state in system services and replays them on the target.
- Checkpoint/Restore in Android (CRIA) transitions an app into a state in which device-specific information can be safely discarded before checkpointing and restoring the app.

Strong points

- Their proposal will affect the development of the mobile sector.
- Their design has contributed to minimizing the penetration of existing mobile OS stacks and applications.

Weak points / Limitations

- If the context never goes away and apps expect it to remain, they may not use conditional reinitialization relied upon by Flux.
- Apps that request to be run in multiple processes are currently unsupported by Flux.

Questions

- It seems to be similar to the function of Samsung Dex. Is Samsung Dex using the suggestions of these authors??

New ideas / Comments

- If users are running a running app on a cloud server, I think the user can solve the problem with a new idea that uses the way that other apps run through the cloud server. With the commercialization of 5G, cloud services can be developed with very low latency. Therefore, we can solve the problem presented in this paper through the cloud service now.