Android Gadgets Apps Scheduling Across Local And Cloud

Priyanka savalagi[1], Pooja Daware[2], Aishwarya patil[3], Vaibhavi Deo[4],Jayashree V. Bagade[5]
[1234]*Student, Department of Information Technology, Vishwakarma Institute of Information Technology, SavitribaiPhule Pune University, Pune, Maharashtra, India*
[5]*Professor, Department of Information Technology, Vishwakarma Institute of Information Technology, SavitribaiPhule Pune University, Pune, Maharashtra, India*

Abstract- We propose a multi-layer mobile application (app) scheduling method to extend the capability of low-end Android devices. We try to reduce this dilemma by a Multi-layer App Scheduling (MAS) schema, along with a cloud service. For the first layer, we utilize the "freeze" feature of Android to prevent non-essential background activities. For the second layer, it is a network scheduler, which automatically schedules the available apps, together with their data, between local and cloud according to user's personal policy generated by big data analysis. By dynamically scheduling the apps among three states, Quality of experience of a low-end Android device is improved. At the same time, with the help of an app state recovery mechanism, the user can directly access a large number of apps provided by the cloud with consistent app view.

Keywords- App Scheduling, Low-End Device, Mobile Device, Android,Cloud backend Platform, Multi-layerApplication Scheduling.

I. INTRODUCTION

The past decade has witnessed the great success of smart devices, such as Android and iOS devices, Besides the mainstream consumer market, there still is a vast market of low-price devices, especially low-end Android devices, in less developed areas. The available resources of those low-end Android devices are constrained either by price or by power capacity multiple running apps may deteriorate Quality of Experience of the user. Moreover, this contradiction between the user requirementsand the limited resources is even severe for low-end devices.On the other hand, the people's requirements of function and performance of end devices are promoted by the rapid development of mobile technology. Therefore we propose a app scheduling schema to extend the capability of android device. devices. An app, together with its user profile and data, can transit among three states.

For example,

1) When the scheduler predicts that the user may use an app with high probability, it reloads the app from the cloud to the local device in advance.
2) If the app is not frequently-used, the scheduler may freeze or
3) Defrost it on demand.
4) If the storage is running out, the scheduler can offload less important apps to the cloud.

1. Background

A great deal of researches have been carried out to extendthe scalability and QoE of smart devices, where some arerelated to this paper. The edge computing technology putsforward many ways to solve the service cache problemof edge nodes, so as to provide services more accuratelyand quickly to IoT devices[3]. The mobile computingoffloading technology uses the terminal monitoring programto monitor the performance of terminals [5]. Themonitoring & reminding mechanism is an easy way andthe mainstream in the software market, and the one of therepresentatives is 360 Mobile Assistant [2]. The small applettechnology such as Android Instant app and WeChatapplet framework , implement a "come and go"mode to run apps that allows users to use apps as loadinga web page, without having to install. Besides, iOS 11 comeswith a new feature named "Offload Unused apps" at WWDC. It lets users free up space for more urgent needs,while still maintaining one-click access to the offloaded apps.The transparent computing technology is also a hottopic in recent years. It is originally a way to solve theperformance problems of traditional terminals, but can alsobe applied to mobile devices. TCID could improve device scalabilityby efficiently querying and effectively loading applicationdata from the network at runtime.

2. Motivation

System propose to boost the demand of low-capability smart devices and also to improve the performance of smart devices, QOE of user. Moreover, to develop storage extension scheme for low-end smart phones.

3. Problem Definition and Objective

Still demand of low-capability smart devices, which are constrained by either capacity or limited resource capability it leads to deteriorate Quality of Experience of the user.

1. To develop a Multi-layer App Scheduling (MAS) schema along with a cloud service.
2. To improve Quality of Experience of a low-end Android device.

3. To provide Storage extension scheme for low end android devices.
4. User can directly access a huge number of apps provided by the cloud.

4. Project Scope and Limitation

The Android gadgets scheduling between local and cloud system aims at helping the users to improve quality of experience of low end android devices. The major benefits of this system are To boost the demand of low-capability smart devices and also to improve the performance of smart phone devices. To develop a storage extension scheme for low-end smartphones.

II. LITERATURE SURVEY

[1] "The lru-k page replacement algorithm for database disk buffering".

This paper introduces a new approach to database disk buffering, called the LRU-K method. The basic idea of LRU-K is to keep track of the times of the last K references to popular database pages, using this information to statistically estimate the interarrival times of references on a page by page basis. In fact, LRU-K an approach the behavior of buffering algorithms in which page sets with known access frequencies are manually assigned to different buffer pools of specifically tuned sizes.

[2] "Dynamically partitioning applications between weak devices and clouds".

This is thefirst ACM Workshop on Mobile Cloud Computing & Services: Social Networks and Beyond Mobile cloud computing applications run diverse workloads under diverse device platforms, networks, and clouds. It introduces the notion of dynamic partitioning of applications between weak devices and clouds and argue that this is key to addressing heterogeneity problems.

[3] "**Dynamic cache cleaning on android. In IEEE International Conference on Communications"**.

This paper introduces caching technique for Android developers cache data to improve the performance of their applications. Caching is the technique of transparently storing also import our contact information into cloud and extract it whenever required. Though it is available on cloud we can access it on any platform or device without any data such that future requests can be accessed more quickly. At times when a mobile device is not under heavy use the cached data, including sensitive data, can remain on the device for an extensive period of time.

[4] "How to back up your iphone, ipad, and ipod touch. https://support.apple.com/en-us/HT203977".

This is apple support site link which tells about different methods of backing up device data into icloud. To let iCloud automatically back up your device each day.With a Wi-Fi network connection, you can make a backup of your device using iCloud. You don't need to plug your device into a computer or even be at home to back up with iCloud. iCloud backups include nearly all data and settings stored on your device.

[5] "Cipher text - Policy Attribute-Based Encryption: An Expressive, Efficient, and Provably Secure Realization"B. Waters, in Proc. 4th Intl Conf. Practice and Theory in Public Key Cryptography (PKC11), 2011, pp. 53-70.

In this paper three constructions within our framework. Our first system is selectively tested safe under the supposition that we call the parallel exponent Bilinear of e-Hellman (PBDHE), which can be considered a generalization of the BDHE assumption. Our the following two constructs provide performance commercials to achieve testable security respectively under the decisive (slope) Bilinear Diffie-Hellman Exponent and decisional Bilinear Diffie-Hellman Assumptions. In an IBE or HIBE system, Keys and encrypted texts are associated with the same kind of simple object: identity. In an ABE system, encrypted keys, and texts are associated with more complex objects: attributes and access to formulas.

III. EXISTING SYSTEM

The existing system focus on delivering high quality multimedia services that can guarantee the agreed QoS and save energy on the mobile devices to increase their lifetime. The current methodologies for vitality sparing in cloud data center concentrate on scheduling occupations between processing servers and giving vitality proficiency by methods for some equipment strategies. For example, we can easily transfer the data as user contents in the form of video, audio, portable document format, text files etc., into the cloud most commonly used in the low end devices is the google drive. It can store chat backup history in most of the apps, we can constraint on exportation. It also suggest you to remove or uninstall the particular application which is seldom use by with the help of data analysis.

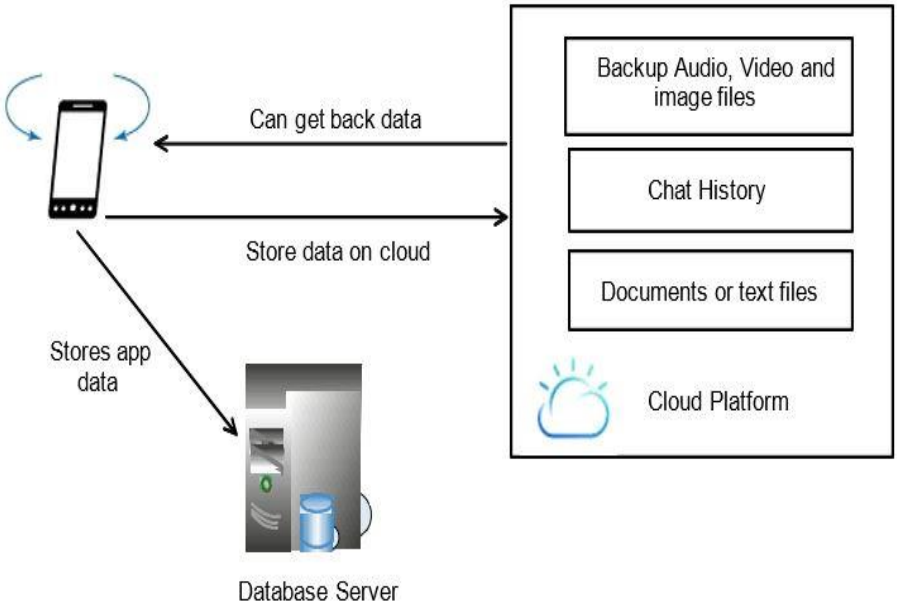


Fig.1: Existing System

Disadvantages of Existing System

1) It can remove application by analyzing but cannot restore the application.

IV. PROPOSED SYSTEM

The propose multilevel app scheduling system consists of three parts: a local app manager (LA Manager), an application level network scheduler (ALN Scheduler), and a Backend Platform in the cloud. LA Manager and ALN Scheduler are build on top of the original Android system without modification, communicating with Backend Platform in the cloud side with remote APIs. LA Manager is the local app manager, consisting of threecomponents: a launcher UI, an auxiliary processing service and a cache. The launcher UI is the home page deployed in the local device to provide a when press and hold an app. The auxiliary processing service follows the instructions from ALN Scheduler to perform auxiliary works before or after any app management action. Main services of it includes Fetch (or push) the backed app

Advantages of Proposed System:

1)Improve Quality of Experience of local devices with the help of cloud computing.
2)Resulting in a better performance of the device.
2)Doesn't provide mechanism to schedule application data and installation file on cloud platform.
3)No mechanism to provide backup of data on cross platform.

unified app access entry. For offloaded apps, a small cue is covered on top of the original app icon to indicate its location. Users can choose an app from the app list to run, no matter it is in the local or in the cloud. If the app is deployed locally, it will start up directly. If the app is in the cloud, it will send a request to ALN Scheduler to start an app reloading process immediately. ALN Scheduleris the general app scheduling manager, who executes app management actions by calling system tools and auxiliary processing service provided by LA Manager. Users can also set the app display policy and the app execution policy by a pop-out menu installation file from (or to) the cloud, and send response to ALN Scheduler when the downloading (or uploading) process is completed for app offloading (or reloading).

3)The user no longer has to uninstall apps due to storage budget.
4)Storage is extended with the help of cloud computing platform.

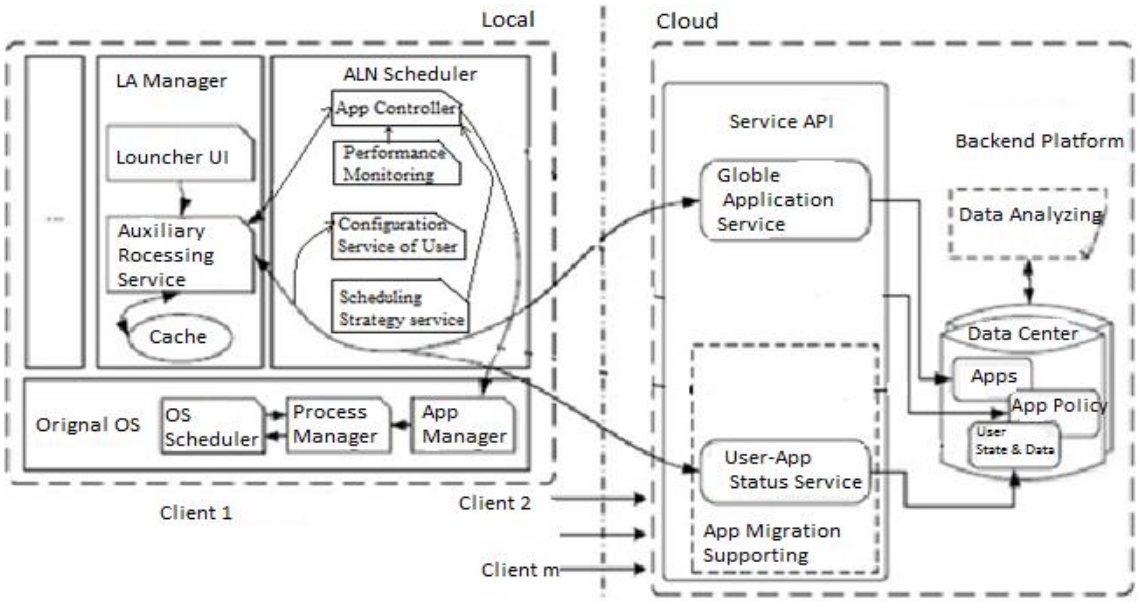
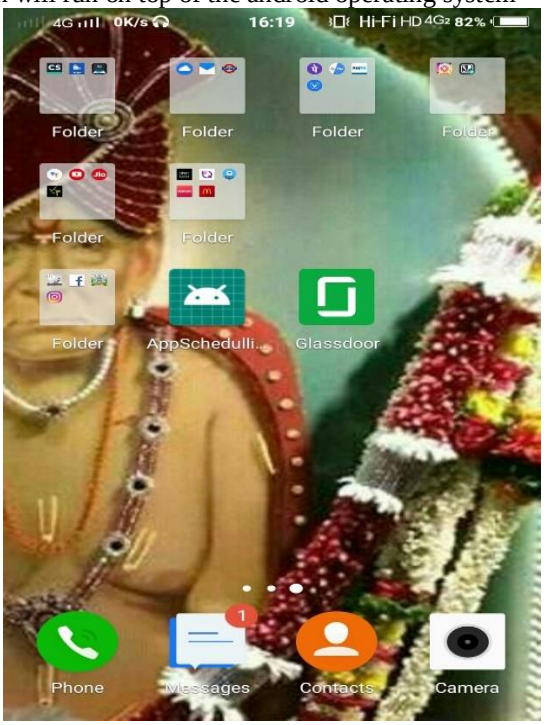


Fig.2: System Architecture

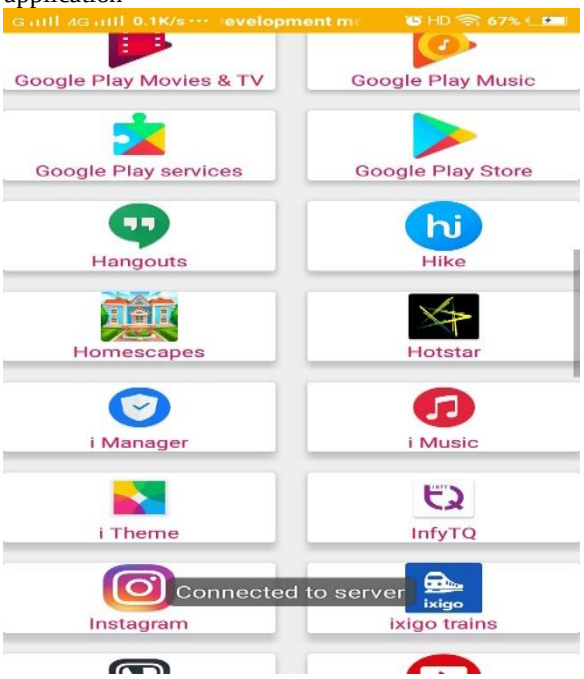
V. RESULT

1) Mobile Screen prompt allow us to select the launcher from the list.

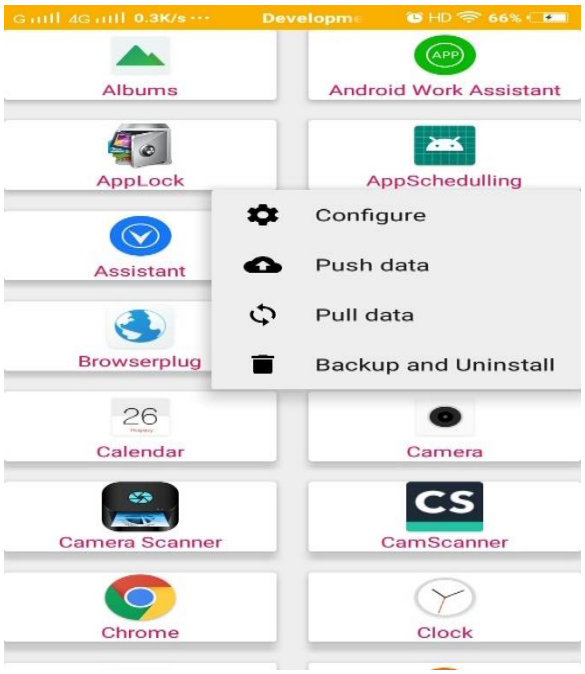
Which will run on top of the android operating system



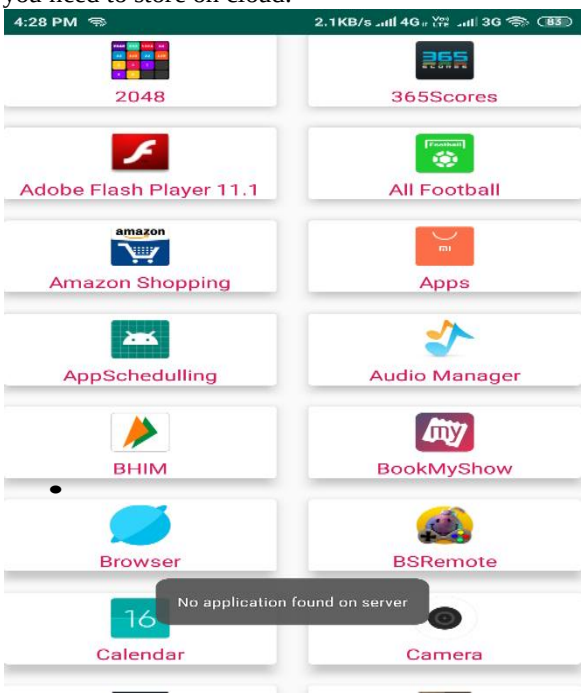
2) When you open application it gets connected to server within couple of seconds and display a message "Connected to server".And Enables user to access the application



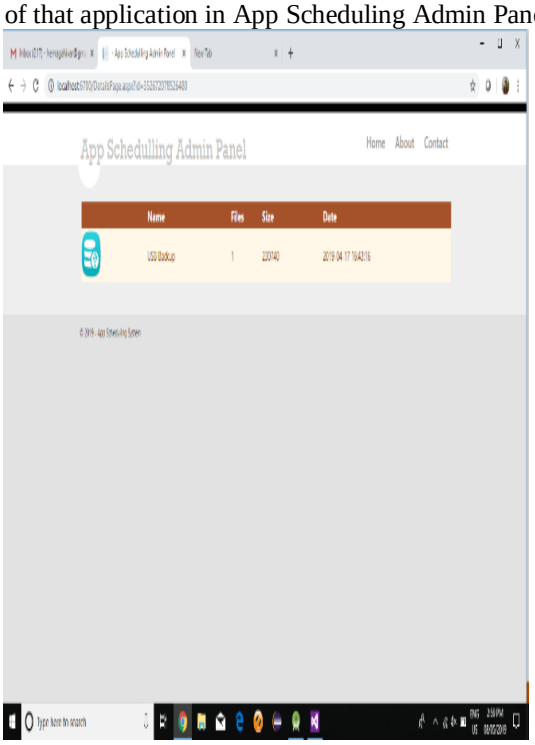
3) After opening application it shows all application of mobile in grid view after pressing the selected application it display settings as configure,push,pull and backup and uninstall.



4) After clicking on push button if configurations settings are not done it displays a message "..." else it allows to push the application data to cloud by selecting what data you need to store on cloud.



5) When user pushes data to cloud it displays user imei number, add application icon which is pushed and files size of that application in App Scheduling Admin Panel



VI. CONCLUSION

We propose a multi-layer app scheduling schema to extend the capability of low-end Android devices. We find that the increasing number of installed apps may degrade Quality of experience of the user, which is even worse on low-end devices. We propose to improve execution time of local devices with the help of cloud computing. With the help of the powerful Backend Platform in the cloud, Quality of a low-end Android device can be enhanced by proactively scheduling apps based on the analysis of user behaviour, device characteristics, etc. At the same time, the user is not needed to be aware of the apps' status, and can directly execute any app provided by the cloud.

VII. APPLICATIONS

1. Can be used and accessible in any Android device with lower or latest versions.
2. Especially can used in anydevice with a low storage capacity and without flexibility to increase storage space of the device.
3. Useful for mobile users in less developed areas who uses low end smartphone device.

VIII. REFERENCES

[1]. K. Akherfi, M. Gerndt, and H. Harroud. Mobile cloud computing forcomputation offloading: Issues and challenges. Applied Computing &Informatics, 2016.

[2]. Testing ui performance. https://developer.android.com/training/testing/performance.html

[3]. E. J. O'Neil, P. E. O'Neil, and G. Weikum. The lru-k page replacementalgorithm for database disk buffering. ACM SIGMOD Record,Jun 1993.

[4]. Chun, ByungGon, Maniatis, and Petros. Dynamically partitioning applications between weak devices and clouds. In ACM Workshop on MobileCloud Computing & Services: Social Networks and Beyond, pages 1–5, 2010.

[5]. S. Finley and X. Du. Dynamic cache cleaning on android. In IEEE International Conference on Communications, pages 6143–6147, 2013.

[6]. Y. Gao,W. Dong, H. Huang, J. Bu, C. Chen, M. Xia, and X. Liu. Whom to blame? automatic diagnosis of performance bottlenecks on smartphones.IEEE Transactions on Mobile Computing, 16(6):1773–1785, Jun 2017.

[7]. L. Yi, J. Li, and Y. Zhang. Improving the scalability of wearabledevices via transparent computing technology. Computing in Science &Engineering, 19(1):29–37, Jan 2017.

[8]. Yi guan chi fan index. http://zhishu.analysys.cn/public/view/wTopApp/wTopApp.html. Accessed Apr 23, 2018.

[9]. L. Zhang. Power, Performance Modeling and Optimization for Mobile System and Applications. PhD thesis, The University of Michigan, 2013.

[10]. Y. Zhang, K. Guo, J. Ren, Y. Zhou, J. Wang, and J. Chen. Transparentcomputing: A promising network computing paradigm. Computing inScience & Engineering, 19(1):7–20, Jan 2017.

[11]. Android instant apps. https://developer.android.com/topic/instant-apps/index.html. Accessed Nov 14, 2017.

[12]. H. Khalid, E. Shihab, M. Nagappan, and A. E. Hassan. What do mobileapp users complain about? IEEE Software, 32(3):70–77, May 2015.