Vol.2011-MBL-60 No.8
Vol.2011-ITS-47 No.8
2011/11/10

Regular Paper

Mobile Wireless Devices, Virtualization and Power Saving

Ahmad Bazzi,[†1] Yoshikuni Onozato[†1] and Kazushi Shinomiya[†1]

Hand-held wireless devices and in particular mobile phones are increasingly gaining new features. Unfortunately the battery technology has not been able to cope with this burst in functions, which create an additional battery load. The research community has been concentrating on improving the power saving in the device by optimizing the different components and technologies within the device. However, the advance of the computer virtualization technologies has lead not only to the emergence of cloud services but also to the birth of mobile virtualization. The use of cloud services and mobile virtualization for power saving is starting to gain momentum with recent research conducted in this field. We believe that this can play an important role in solving the power limitation problem.

1. Introduction

Mobile devices are continuously getting richer in features. The mobile phone that once allowed only voice calls and basic text messages has grown now to run all kinds of applications, from web browsing to GPS navigation. These new features created new possibilities and allowed the users to enjoy the convenience of new functions; however, this also comes at a cost represented by a shorter battery life. A discharged battery means that the mobile phone is no longer capable of maintaining its most basic function: phone call. This of course extends beyond just mobile phones to encompass tablets and even portable audio players. Many manufacturers are designing their tablets to support many wireless protocols, such as WiFi 802.11 and Bluetooth. Even portable audio players are now expected to browse the Internet, run applications and even function as a basic personal navigation system thanks to the built-in GPS.

†1 Gunma University

The increase in feature-packing and the boom in mobile applications indicate that the battery power is in high demand and is more precious than ever. The design of these portable devices has to compromise between portability and battery size and capacity. From the user's point of view, this means that she will need to charge her device more often and any excessive activity can risk draining her battery charge and eventually losing the most basic functions of the device, such as phone calls and messaging. From the manufacturer's point of view, it becomes vital to make utmost use of the available battery power by reducing the power consumption of the different components, such as the display, the processor and the network cards.

Computer virtualization has laid the foundation for cloud services[1]), which allow the clients to achieve a new level of scalability and processing power. On the other hand, it inspired porting virtualization to the ARM processor, giving birth to mobile virtualization[2]). Cloud services and mobile virtualization can be utilized to save power on mobile devices, for example by using a server to run a computationally intensive task instead of running it on the battery-powered hand-held device.

In this paper we are interested in how virtualization can be used for mobile device power saving whether directly through mobile virtualization or indirectly through cloud services. The rest of this paper is organized as follows, in Section 2, we describe computer virtualization technology. In Section 3 and Section 4, we discuss mobile virtualization applications and how they can contribute to power saving respectively. We summarize cloud services in Section 5, and discuss the potential power saving techniques in Section 6. In Section 7, we review the previous works related to the network interface card power saving. Finally we conclude in Section 8.

2. Computer Virtualization

Computer virtualization originally started with IBM in the 1960s and continued till the 1970s[3]). It is a technology that allows running more than one OS on a single computer. The virtualization of the different servers and appliances has become quite feasible using any of the available free hypervisors and virtualization programs. This is achieved by creating a virtual hardware layer for the guest OS to run, and hence the term virtualization.

During the 1980s, the interest in virtualization has decreased as the personal computer became more popular on the account of the mainframes. However, this interest has risen again in the last decade especially after VMware managed to successfully virtualize the x86 processor[4] and released their first x86 virtualization product in 1999. Recently, not only servers, but also desktop computers and many laptops are shipped with multi-core processors. This is creating plenty of CPU processing power to push virtualization support even further. Coupled with the spread of multi-core CPUs, virtualization is gaining more power than ever.

Moreover, in the past years, virtualization has become strongly endorsed by the different operating system vendors and CPU manufacturers. For example, Red Hat Linux was shipped with built-in virtualization support from Xen and now with KVM[⋆1][5]. VMware offers VMware Player and VMware vSphere Hypervisor (ESXi) with basic management tools for free while offering more advanced functions for a non-free license[6]. Microsoft is also offering virtualization support in their newer operating systems and they even released Microsoft Hyper-V Server, a hypervisor, for free[7]. With this number of products, it is becoming more accessible for everyone from individuals to universities and companies to test and implement virtualization technologies.

On the CPU manufacturer's side, AMD has added hardware-assisted virtualization to their newer processors using AMD-V[8], while Intel is equipping its new processors with Intel VT[9].

2.1 Virtualization Layers

Computer virtualization is a technology where a software layer lies directly or indirectly on top of the hardware to allow multiple guest operating systems to run in virtual machines (VMs). A VM will appear like a real computer from the guest operating system perspective, while it is only an abstraction of the underlying computer resources[10].

Fig. 1 shows the different components of a virtual system. At the base level, we have the physical hardware that is the computer hardware. In type 1 hypervisor, the virtualization software runs directly on top of the physical hardware instead of the OS. In type 2 hypervisor, the virtualization software runs on top of the host OS along with the other applications. Moreover the hypervisor is responsible for creating a layer of virtual hardware to run the guest operating system(s).

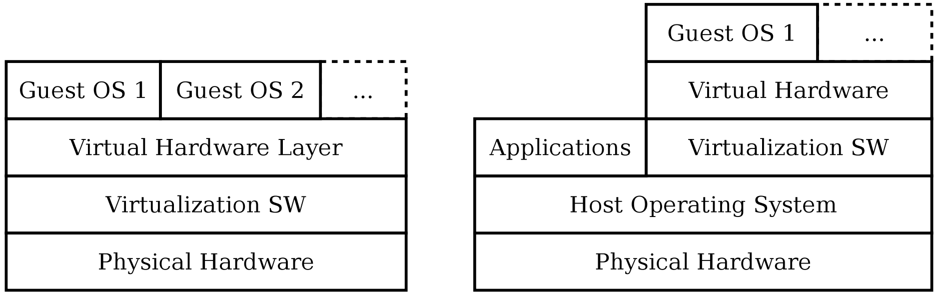

Fig. 1 Virtualization system common types. In full virtualization, the VMM can run either directly on top the physical hardware or on top of the host OS. The VMM creates a layer of virtual hardware for the guest OS to run.

The virtualization software has a fundamental component, the Virtual Machine Monitor (VMM), which provides us with the ability to run VMs. The VM appears and functions as a real computer as it interacts with the guest OS, which can be installed and run on it. Hence, virtualization allows us to run more than one operating system on the same hardware where each guest OS can have its private independent virtual hardware.

2.2 Virtualization Applications

Initially, the purpose of computer virtualization was to run more than one OS on a single computer. Knowing that computer resources are generally underutilized, virtualization can allow better use of the available hardware. Now, different innovative uses have arisen, such as live migration[11)–12] and higher up-time.

Now, new technologies are being developed that allow new possibilities that are not possible without relying on virtualization. For example,

Live Migration This technique allows moving a running operating system from one physical host to another with negligible or no downtime.

Snapshots This technique allows saving the state of a virtual machine, which can be running or turned off, and restoring that state at any point in the future.

⋆1 http://www.linux-kvm.org/

However, virtualization is now leaving the desktop and server to encompass mobile phones. Some research is interested in virtualization on mobile phones in order to achieve added security.

3. Virtualization on Mobile Phones

In addition to computer virtualization, there is now an ongoing research to use virtualization on mobile devices as well. For published mobile virtualization literature, we refer the reader to VMware Mobile Virtualization Platform[13)], MobiVMM[14)] and VIRTUS[15)]. On the other hand, there are several companies who are now offering commercial products and services built upon mobile virtualization. Examples include B Labs⋆1, Open Kernel Labs⋆2 and Red Bend⋆3, who have their current service and product lines based on mobile virtualization.

Mobile virtualization seems to be motivated by the quest for corporate security that should encompass the mobile phones through which corporate data are being accessed and transmitted[13)]. On a more general scale, Brakensiek et al. argues that virtualization is an "enabler for security in mobile devices[16)]." Furthermore, mobile virtualization is now creating new levels of services that were not feasible before. Finally, the concept of a virtual phone that can be moved from a hypervisor to another is creating an interest for pervasive computing researchers[17)].

3.1 Mobile Virtualization and Security

Corporate security policy need to make sure that all the devices that connect to the corporate network must follow strict security policies. The security threats is not limited to malware and viruses in the traditional sense, but extends beyond that to include mobile applications that collect and send private data, such as device ID, location and even contacts[18)].

One solution that the companies have adopted is giving the employees one phone for corporate use that would allow them to access the company services. The applications installed on the corporate mobile are limited to those approved by the company's IT department in order to ensure data security. Consequently an employee would end up carrying two phones one for corporate use and another for his personal use so that she can install her favorite applications.

Mobile virtualization tackles this problem by allowing one physical phone to run two —or more— virtual phones. This allows a company IT department to provision and control the corporate virtual phone, while the employee remains free to install any application she wants on her personal virtual phone. This solution gives the employee the freedom to use her personal phone to access corporate resources and allows the "Bring Your Own Device" (BYOD) approach.

3.2 VMware Mobile Virtualization Platform

One example of virtualization on mobile phones is the VMware Mobile Virtualization Platform (MVP)[13)]. MVP is designed as a type-2 (hosted) hypervisor. It installs as an application on the mobile phone operating system and runs alongside other applications; however, as a hypervisor, it can provide the ability to run guest virtual machines. Hence, this hypervisor does not need to be integrated early on by the phone manufacturer.

One of the design goals of this project is to offer the security while allowing the BYOD model in companies. An employee can bring his own mobile device that runs his own set of personal applications, and use it as his work phone without compromising the company's security policies. This is achieved by running a company provisioned image, i.e. a virtual phone image, on the user's phone. This virtual phone will adhere to the company's security policy, such as "minimum password strengths, inactivity thresholds and lockout, storage and network encryption and remote wipe"[13)], thus ensuring that the company's security requirements are met.

On the other hand, by ensuring isolation between the corporate virtual phone from the user's applications, the user will continue to install any applications he likes while the company is still able to protect its corporate data.

3.3 New Services

Mobile virtualization created a whole new set of services that were not possible before. In his article that discusses the evolution of virtualization, Kroeker presents the various opinions of key persons in virtualization industry from VMware, Citrix and AMD[19)]. Although they might not hold the same vision, we can see that the interest in bringing virtualization to mobile phones can have the following advantages[19)]:

⋆1 http://b-labs.com/
⋆2 http://www.ok-labs.com/
⋆3 http://www.redbend.com/

"Test once, deploy on different devices" This can save the mobile device manufacturer a lot of work if the phone OS were going to run on identical virtual hardware (hypervisor).

New levels of backup and restore The mobile carrier would be able to offer a new level of backup/restore to new handsets by knowing that the underlying hypervisors are compatible. In other words, the backup/restore can be done on the virtual machine level.

Two virtual phones on a single physical one This would allow the employees to dedicate on virtual phone for their business communication and requirements, while have the other virtual phone for their personal use. This can increase the security because the underlying virtualization technology can keep the two virtual instances isolated.

Compartmentalization Compartmentalization means dedicating a VM for every component. The handset can incorporate a "virtual machine that controls the radio, another for the default software and applications, and a third that operates everything the user downloads and installs."

4. Mobile Virtualization and Power Saving

The current virtualization technology and products allows the live migration of a virtual machine from one system to another. Mobile Internet Devices (MID) generally run on limited battery power and have relatively low processing power. So it is quite convenient for the user to migrate her current system from a battery-powered device to a desktop whenever possible as this would let her benefit from faster processing while saving battery power. Sud et al. researched the case of using live migration to offload computation from a MID to a desktop[20]. In particular they used KVM for dynamic computation offload from a netbook running Intel Atom processor to a laptop on the same LAN[20]. Their experiment showed that the migration can take up to 25 seconds over a 100 Mbps Ethernet network and up to 150 seconds over an 802.11 Draft N capable of 70 Mbps data transfer.

As the phone is virtualized to run on a hypervisor, it becomes possible to transfer this virtual phone to a different device that is not battery powered. Live migration from an Intel Atom-based device to a desktop is already feasible with the available software solutions[20]; however, on a more general scale, computation migration from a mobile phone —usually using an ARM processor— to another device with higher processing power is only still being researched. The mobile virtualization model allows running multiple virtual phones on a single physical device. One of the virtual phones might be dedicated to run processing demanding applications, such as video editing or image/face recognition. Rudolph suggests that the new host device can be any computational or digital device that can run the hypervisor, and this goes far beyond just hand-held devices and computers[17]. This approach allows running more computationally demanding applications without being limited by the phone battery or even the phone processing capabilities.

On a different level, the virtual phone[13] can be migrated to a dedicated server —such as a cloud-hosted hypervisor— to finish its tasks without draining the local device's limited resources.

5. Cloud-Based Services

The development of virtualization technologies has given birth to cloud computing. Cloud computing can deliver three kinds of services[21],

Infrastructure as a Service (IaaS) IaaS allows the client to access directly the underlying virtual infrastructure in order to create and run his virtual machines.

Platform as a Service (PaaS) PaaS gives the client access to an operating system to setup applications and configure them as he sees fit.

Software as a Service (SaaS) SaaS gives the client direct access to applications that he can use directly without being concerned with the setup details.

The available high-speed wireless network connections allow the user to take advantage of the cloud services on his wireless device. Clients mainly need storage and applications, and they can accomplish that by utilizing cloud services. The cloud provider is responsible for the setup, maintenance, security and backup of the servers in its data center.

This creates new potentials that were not possible earlier with hand-held devices. In particular, we mention:

- (Virtually) Unlimited Storage: By making use of the cloud storage, a client can access data beyond the physical capacity of any mobile device, as long as there is network access, irrespective of place.

- (Virtually) Infinite Processing Power: Cloud hosted servers can offer a huge processing power that can be utilized even by mobile devices.

5.1 Cloud Storage

Cloud storage allows practically unlimited storage space for the users. With a proper subscription, the user can have access to a storage space that far surpasses the physical storage of his device. At the time of writing, it is common to see a 100 GB cloud storage versus a 16 GB internal storage in a hand-held device. Consequently this approach is replacing the need to constantly carry one's data —such as music or videos— on his portable device and instead access them when needed. In other words, users can now keep the limited internal storage for the files that they access on a regular basis, and rely on cloud storage to save the less accessed files to download them on a need-to-basis.

From the device manufacturer point of view, the need to increase the internal storage can become less pressing. Considering the cost of internal storage, this approach can keep the wireless hand-held devices more affordable, yet without limiting the device usage possibilities.

Sharing Services

One cloud service that is becoming increasingly popular is file sharing. Without the need to resort to e-mail, users can easily share files between each other simply by using a cloud-based storage. This can be of great utility especially between friends or groups working on the same project.

5.2 Cloud-hosted Applications

The rising trend to host applications on the cloud —that can be instantly accessed by any wireless device— has created endless possibilities. The application running on a mobile device is no longer bounded by the processing power of the latter. This allowed us to achieve new levels in data creation, manipulation and sharing that would not have been possible if we were to rely on the internal processor alone. The user can now access and enjoy a vast computing power even from his limited-battery-powered device.

Gradually the demand for faster processors can decline, allowing the manufacturer to shift more emphasis to other components. This can also help cope with the battery saving challenge; generally speaking a "slower" processor consumes less power allowing the device battery to run for a longer period of time.

6. Mobile Power Saving and the Cloud

The migration of certain computing-intensive tasks to the cloud means that we are trading bandwidth for processing power and consequently battery life.

One prototype design for using the cloud computing resources to support in running local programs is presented by Chun et al. in CloneCloud[22]. CloneCloud targets application-layer virtual machines such as Java and MS .Net. It is based on the idea of migrating a process from the mobile device to a clone in the cloud, execute it on that server, and reintegrate it back when it finishes. Following the objective to optimize the execution time and energy, the process is migrated to the cloud or run locally. CloneCloud was able to achieve up to 20× reduction in energy and 20× reduction in execution time.

6.1 Offloading Computation Decision

Deciding whether to offload computing requires forecasting the cost of running the process locally versus uploading it to a remote server. The cost can be in terms of computation time[23] or in terms of energy[24].

From the time perspective, Wolski et al. present a framework to evaluate the cost of running the process locally versus offloading it to another server. Expectedly, the local cost would be the time required for computing locally, while the remote cost would include the sending/receiving transmission time to/from the remote system in addition to the time required for computing remotely[23].

However, in this paper, we are more concerned with the energy point of view. Kumar and Lu discuss some points to consider in order to decide whether offloading computation can save power. In brief, the amount of energy saved is[24],

$$Power\ Saving = P_c \times \frac{C}{M} - P_i \times \frac{C}{S} - P_{tr} \times \frac{D}{B} \tag{1}$$

where,
P_c is the power that the mobile system consumes for computing
P_i is the power that the mobile system consumes for being idle
P_{tr} is the power that the mobile system consumes for sending and receiving data
C is the number of instructions required for the computation

M is the speed of the mobile system
S is the speed of the cloud server
D is the number of bytes to be transferred
B is the bandwidth

In this case, offloading computation to a cloud server will be beneficial as long as *Power Saving* yields a positive value.

6.2 Cloud-hosted Anti-virus

In addition to offloading computation on a need-to basis, certain services can be permanently moved to the cloud. In particular, one model was proposed and tested where the antivirus functionality is moved to a remote network service that employs multiple virtualized malware-detection engines[25]. According to Oberheide et al., this model has three main advantages[25],

(1) Better detection of malicious software: By moving the anti-virus functionality to a remote server, it becomes more feasible to provide all the resources required for more thorough virus detection.

(2) Decreased utilization of resources: Instead of running an anti-virus and scanning untrusted files locally, sending the files to a remote server for inspection leads to a lower utilization of local resources.

(3) Reduced software complexity: The authors for instance were able to achieve their program by writing 170 lines of Python code; this is in clear contrast with the complicated design of any antivirus application.

It is worth noting that in a different publication, Oberheide et al. introduce N-version anti-virus which is an anti-virus as a cloud service that utilizes multiple heterogeneous detection engines[26]. At the time of writing, there is at least one company offering a cloud-based anti-virus for computers[27].

6.3 Challenges and Obstacles

Adopting cloud services can face several obstacles[1], for example:

Data Confidentiality By moving the user's data to the cloud, the user's privacy and data security becomes strictly dependent on the cloud provider. Any bug or vulnerability in the provider's system might expose the user's data.

Availability of a Service Using cloud services requires that the client is connected to the network, otherwise, access to the service is lost. On the other hand, if the cloud provider is suffering a service outage or overload, then similarly —from the client's point of view— the access to the service is lost.

7. Power Saving Mode

Both cloud services and mobile virtualization through computation offloading requires network access to connect with other servers or hosts. As a result, power saving through the reliance on these technologies creates more emphasis on optimizing power saving in network interface cards (NICs), such as the commonly used IEEE 802.11.

In the past years, there have been various works attempting to improve the power saving of 802.11 wireless networks, mainly through improving the 802.11 Power Saving Mode (PSM). We can see that the research trend covers both PSM modifications based on mathematical algorithms and modifications based on traffic-aware systems.

In 2004, Nath et al. proposed a Dynamic Beacon Period (DBP)[28], where the access point uses different beacon periods for the different clients. The beacon period is calculated by the client based on the RTT of the client's current HTTP connection. Once chosen, it notifies the AP of its choice and the AP in turn uses this beacon period to communicate with the client. Hence this is different from the standard case where the same beacon period is used uniformly across all the clients.

Krashinsky and Balakrishnan proposed Bounded-Slowdown (BSD) protocol[29]. BSD protocol aims to minimize energy consumption without letting the round trip time (RTT) exceed a certain configurable limit, eventually a compromise can be reached between power saving and speed. This protocol relies on past network activity in order to adapt the sleep duration, while ensuring that the RTT is not stretched by more than a chosen factor p. Consequently, the network interface will wake up less often to listen for the beacon during idle times. The authors provide different simulations when BSD is used in the case of HTTP requests issued to a web server.

Qiao and Shin proposed Smart Power Saving Mode (SPSM)[30], which aims to find the best action sequence to enter the wake and doze states in order to guarantee a limited delay with minimum energy consumption. An action sequence is what tells the network interface to stay awake or sleep after a request is sent in. In order to achieve this optimal action sequence, SPSM uses a two-step approach. First, it interprets the desired delay performance using a generic penalty function. Second, it feeds this penalty function to their proposed algorithm in order to produce the optimal action sequence. According to their simulations, SPSM provided and ensured the same responsiveness as BSD while achieving additional energy saving.

On the other hand, we observe PSM modifications that operate from the application level. Anand et al. proposed a self-tuning power manager (STPM)[31]. In order for this approach to work, each application has to provide *hints* regarding when it wants to use the wireless network. These hints show when this application is transferring data and the maximum delay that it tolerates. Consequently, applications should be modified in order to incorporate this new feature; however, the authors also considered "legacy" applications. In this case, a module can be used to identify the non-hinting applications, observe their traffic and issue hints on their behalf. The authors implemented STPM within the operating system, as a Linux kernel module, and ran several tests. They were able to reduce the power consumption while decreasing the delay by up to 80%.

Anastasi et al. presented a Cross-Layer Energy Manager (XEM)[32] which implements a mixed policy. In this new policy, the wireless interface is switched between PSM and off mode depending on the traffic generated by the respective host. This approach requires an agent that is aware of the specific application network utilization pattern to be running on the host. Eventually, in one version of XEM a customized energy manager is required for each network application.

Finally, we should mention the research concerned with saving power by modifying the AP. Lee et al. suggested tuning the beacon period (BP)[33] in order to achieve better energy saving and minimize delay. The authors propose two heuristic scheduling policies aiming to save energy.

Xie et al. proposed an AP-centric PSM[34], where the objective is to have the AP compute the optimal PSM parameters, i.e. the beacon interval, listen interval and congestion window size. They proposed an algorithm to achieve this and simulated it to study the performance.

Ogawa et al. considered the case where the AP can sleep as in battery-powered cases and they introduce a new action frame —power saving access point (PSAP) action frame— to inform the connected nodes about the its sleep length. Moreover, these nodes would also enter in a sleep state after receiving the PSAP action frame[35].

Live migration of mobile virtual machines and cloud services require heavy usage of the wireless networks. Accordingly optimizing the power saving in NICs is equally important in order for these technologies to be successful.

8. Conclusion

The research in the past years has been concentrating on improving the power saving on hand-held devices by optimizing the different technologies that lie within the device, such as, LCD display, CPU, network interface card, battery utilization, etc. The advent of virtualization and the birth of the cloud-based services, along with the current high-speed networks, have created new possibilities that allow migrating power-demanding applications to other servers.

References

1) Armbrust, M., Fox, A., Griffith, R., Joseph, A.D., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., Stoica, I. and Zaharia, M.: Above the Clouds: A Berkeley View of Cloud Computing Cloud Computing, *Computing*, pp.07–013 (2009).
2) Cox, L.P. and Chen, P.M.: Pocket Hypervisors: Opportunities and Challenges, *Eighth IEEE Workshop on Mobile Computing Systems and Applications, 2007. HotMobile 2007*, IEEE, pp.46–50 (online), DOI:10.1109/HotMobile.2007.20 (2007).
3) Padegs, A.: System/360 and Beyond, *IBM Journal of Research and Development*, Vol.25, No.5, pp.377–390 (1981).
4) VMware Inc.: Understanding Full Virtualization, Paravirtualization, and Hardware Assist (2007).
5) Red Hat: Red Hat Enterprise Virtualization for Servers (online), available from ⟨http://www.redhat.com/virtualization/rhev/server/⟩ (accessed 2011-10-10).
6) VMware Inc.: VMware vSphere Hypervisor: Bare Metal Hypervisor (online), available from ⟨http://www.vmware.com/products/vsphere-hypervisor/⟩ (accessed 2011-10-10).
7) Microsoft Corporation: Microsoft Hyper-V Server (online), available from ⟨http://www.microsoft.com/hyper-v-server/⟩ (accessed 2011-10-10).
8) Advanced Micro Devices Inc.: AMD-V Nested Paging (2008).

9) Neiger, G., Santoni, A., Leung, F., Rodgers, D. and Uhlig, R.: Intel virtualization technology: Hardware support for efficient processor virtualization, *Intel Technology Journal*, Vol.10, No.3, pp.167–177 (2006).
10) VMware Inc.: Virtualization Overview (2006).
11) Clark, C., Fraser, K., Hand, S., Hansen, J., Jul, E., Limpach, C., Pratt, I. and Warfield, A.: Live migration of virtual machines, *Proceedings of the 2nd conference on Symposium on Networked Systems Design & Implementation-Volume 2*, No. Vmm, USENIX Association, pp.273–286 (2005).
12) VMware Inc.: VMware vMotion for Live Migration of Virtual Machines (online), available from ⟨http://www.vmware.com/products/vmotion/⟩ (accessed 2011-10-10).
13) Barr, K., Bungale, P., Deasy, S., Gyuris, V., Hung, P., Newell, C., Tuch, H. and Zoppis, B.: The VMware mobile virtualization platform: is that a hypervisor in your pocket?, *ACM SIGOPS Operating Systems Review*, Vol.44, No.4, pp.124–135 (2010).
14) Yoo, S., Liu, Y., Hong, C., Yoo, C. and Zhang, Y.: Mobivmm: a virtual machine monitor for mobile phones, *1st International Conference on Mobile System Virtualization (MobiVirt 2008)* (2008).
15) Inoue, H., Ikeno, A., Kondo, M., Sakai, J. and Edahiro, M.: VIRTUS: a new processor virtualization architecture for security-oriented next-generation mobile terminals, *Proceedings of the 43rd annual conference on Design automation - DAC '06*, New York, New York, USA, ACM Press, pp.484–489 (online), DOI:10.1145/1146909.1147038 (2006).
16) Brakensiek, J., Dröge, A., Botteck, M., Härtig, H. and Lackorzynski, A.: Virtualization as an enabler for security in mobile devices, *Proceedings of the 1st workshop on Isolation and integration in embedded systems*, No.April, ACM, pp.17–22 (2008).
17) Rudolph, L.: A virtualization infrastructure that supports pervasive computing, *Pervasive Computing, IEEE*, Vol.8, No.4, pp.8–13 (2009).
18) Thurm, S. and Kane, Y.I.: Your Apps Are Watching You The Wall Street Journal, (online), available from ⟨http://on.wsj.com/eT00xY⟩ (2010).
19) Kroeker, K.: The evolution of virtualization, *Communications of the ACM*, Vol.52, No.3, pp.18–20 (online), DOI:10.1145/1467247 (2009).
20) Sud, S., Want, R., Pering, T., Lyons, K., Rosario, B. and Gong, M.X.: Dynamic Migration of Computation Through Virtualization of the Mobile Platform, *Mobile Networks and* (2010).
21) Fouquet, M., Niedermayer, H. and Carle, G.: Cloud computing for the masses, *Proceedings of the 1st ACM workshop on User-provided networking: challenges and opportunities*, ACM, pp.31–36 (2009).
22) Chun, B.-G., Ihm, S., Maniatis, P., Naik, M. and Patti, A.: CloneCloud, *Proceedings of the sixth conference on Computer systems - EuroSys '11*, New York, New York, USA, ACM Press, p.301 (online), DOI:10.1145/1966445.1966473 (2011).
23) Wolski, R., Gurun, S., Krintz, C. and Nurmi, D.: Using bandwidth data to make computation offloading decisions, *2008 IEEE International Symposium on Parallel and Distributed Processing*, IEEE, pp.1–8 (online), DOI:10.1109/IPDPS.2008.4536215 (2008).
24) Kumar, K.: Cloud Computing for Mobile Users: Can Offloading Computation Save Energy?, *Computer*, Vol.43, No.4, pp.51–56 (online), DOI:10.1109/MC.2010.98 (2010).
25) Oberheide, J., Veeraraghavan, K., Cooke, E., Flinn, J. and Jahanian, F.: Virtualized in-cloud security services for mobile devices, *Proc. Workshop on Virtualization in Mobile Computing*, Citeseer (2008).
26) Oberheide, J., Cooke, E. and Jahanian, F.: CloudAV: N-version antivirus in the network cloud, *Proceedings of the 17th conference on Security symposium*, USENIX Association, pp.91–106 (2008).
27) Panda Security: Panda Cloud Antivirus (online), available from ⟨http://www.cloudantivirus.com/⟩ (accessed 2011-10-10).
28) Nath, S., Anderson, Z. and Seshan, S.: Choosing beacon periods to improve response times for wireless HTTP clients, *Proceedings of the second international workshop on Mobility management & wireless access protocols*, ACM, pp. 43–50 (2004).
29) Krashinsky, R. and Balakrishnan, H.: Minimizing energy for wireless web access with bounded slowdown, *Proceedings of the 8th annual international conference on Mobile computing and networking*, ACM, pp.119–130 (2002).
30) Qiao, D.: Smart power-saving mode for IEEE 802.11 wireless LANs, *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, Vol.3, pp.1573–1583 (2005).
31) Anand, M., Nightingale, E. and Flinn, J.: Self-tuning wireless network power management, *Wireless Networks*, Vol.11, No.4, pp.451–469 (2005).
32) Anastasi, G., Conti, M., Gregori, E. and Passarella, A.: 802.11 power-saving mode for mobile computing in Wi-Fi hotspots: limitations, enhancements and open issues, *Wireless Networks*, Vol.14, No.6, pp.745–768 (2008).
33) Lee, J., Rosenberg, C. and Chong, E.: Energy efficient schedulers in wireless networks: design and optimization, *Mobile Networks and Applications*, Vol.11, No.3, pp.377–389 (2006).
34) Xie, Y., Luo, X. and Chang, R.: Centralized PSM: An AP-centric power saving Mode for 802.11 infrastructure networks, *Sarnoff Symposium, 2009.*, IEEE, pp.1–5 (2009).
35) Ogawa, M., Hiraguri, T., Nishimori, K., Honma, N., Takaya, K. and Murakawa, K.: Power Saving Control Method for Battery-Powered Portable Wireless LAN Access Points in an Overlapping BSS Environment, *IEICE Transactions on Communications*, Vol.94, No.3, pp.658–666 (2011).